Shape Reconstruction from Shadows and Reflections

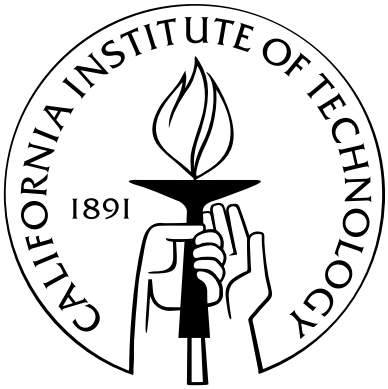
Thesis by

Silvio Savarese

In Partial Fulfillment of the Requirements

for the Degree of

Doctor of Philosophy



California Institute of Technology

Pasadena, California

2005

(Defended)

A mio Padre e mia Madre

Acknowledgements

I wish to thank my advisor, Pietro Perona. His advice and guidance during these years have made a substantial difference in my work. I also wish to express my gratitude to Dr Fausto Berbardini, Professor Holly Rushmeier, Professor Gabriel Taubin, Dr Min Chen and Marco Andreetto for their close collaboration. I am grateful to Professor Jean Ponce, Professor Jerry Marsden, Professor Stefano Soatto, Dr Marzia Polito, Dr Matthew Cook, Dr Jean-Yves Bouguet for helpful feedback and many fruitful discussions. I would also like to thank the other members of my committee, Professors James Arvo, Demetri Psaltis, Shinsuke Shimojo and Jehoshua Bruck.

I wish to thank past and present members of the Caltech vision group for sharing moments of inspiration and fun. A special mention goes to my old friend Massimo. Finally, I wish to thank my parents and Fei-Fei for their amazing encouragement and continuous support.

Abstract

Measuring automatically the shape of physical objects in order to obtain corresponding digital models has become a useful, often indispensable, tool in design, engineering, art conservation, computer graphics, medicine and science. Machine vision has proven to be more appealing than competing technologies. Ideally, we would like to be able to acquire digital models of generic objects by simply walking around the scene, while filming with a handheld camcorder. Thus, one of the main challenges in modern machine vision is to develop algorithms that: i) are inexpensive, fast and accurate; ii) can handle objects with arbitrary appearance properties and shape; and iii) need little or no user intervention.

In this thesis, we address both issues. In the first part, we present a novel 3D reconstruction technique which makes use of minimal and inexpensive equipment. We call this technique ”shadow carving”. We explore the information contained in the shadows that an object casts upon itself. An algorithm is provided that makes use of this information. The algorithm iteratively recovers an estimate of the object which i) approximates the object’s shape more and more closely; and ii) is provably an upper bound to the object’s shape. Shadow carving is the first technique to incorporate ”shadow” information in a multi-view shape recovery framework. We have implemented our approach in a simple table-top system and validated our algorithm by recovering the shape of real objects.

It is well known that vision-based 3D scanning systems handle specular or highly reflective surfaces only poorly. The cause of this deficiency is most likely not intrinsic, but rather due to our lack of understanding of the relevant cues. In the second part of this thesis, we focus on how to promote mirror reflections from ”noise” to ”signal”. We first present a geometrical and algebraic characterization of how a patch of the scene is mapped into an image by a mirror surface of given shape. We then develop solutions to the inverse problem of deriving surface shape from mirror reflections in a single image. We validate our theoretical results with both numerical simulations and experiments with real surfaces.

A third goal of this thesis is advancing our understanding of human perception of shape from reflections. Although the idea of perception of shape from different visual cues (e.g., shading, texture, etc.) has been extensively discussed in the past, little is known to what extent highlights and specular reflections carry useful information for shape perception. We use psychophysics to study this capability. Our goal is to provide a benchmark, as well as inspire possible technical approaches, for our computational work. We find that surprisingly, humans are very poor at judging the shape of mirror surfaces when additional visual cues (i.e., contour, shading, stereo, texture) are not visible.

Contents

Chapter 1

Introduction

1.1 Toward an Ideal Visual System to Acquire Digital Models

For the last thirty years, the possibility of capturing digital representations of physical objects has been receiving an enormous amount of attention in information technology research. Contrary to physical models, digital models can be easily stored, retrieved in databases, edited in computer-aided-design (CAD) based software, transmitted and received through digital communication systems, used as templates for reverse engineering, displayed and analyzed at different resolution, and more. As a result, digital models are used in many applications. Perhaps the most important ones are archaeology, education, medicine, entertainment, virtual reality, computer graphics, design and prototyping of industrial components, e-commerce, and robotics.

There exist several different ways of capturing digital models. Machine vision has proven to be more attractive than many competing technologies. For instance, acquiring models from visual inputs (such as images) is, by far, more convenient and faster than making use of touch probes. Ideally, one would like to be able to acquire digital models by simply walking around the scene, while filming with a handheld camcorder.

This possibility has boosted vision research enormously and several techniques have been proposed. These techniques mainly differ in: i) the type of visual cues being used for shape recovery (texture, shading, contours systems); ii) the number of observers (monocular, stereo or multi-view stereo); and iii) the possibility of projecting ”active cues” such as structured lighting on the object surface (laser stripes, structured lighting patterns, shadows, etc).

However, much work still needs to be done. Each of those solutions present drawbacks. For instance, structured lighting techniques tend to be accurate, but expensive. Also, they require an additional step in which all of the reconstructed portions of the surface need to be stitched together. Reconstruction schemes that use multiple cameras are hard to calibrate. Systems that just use ”passive” cues such as contours are proven to be robust (e.g., multiple views are easily stitched together), but inaccurate. As a result, we are still

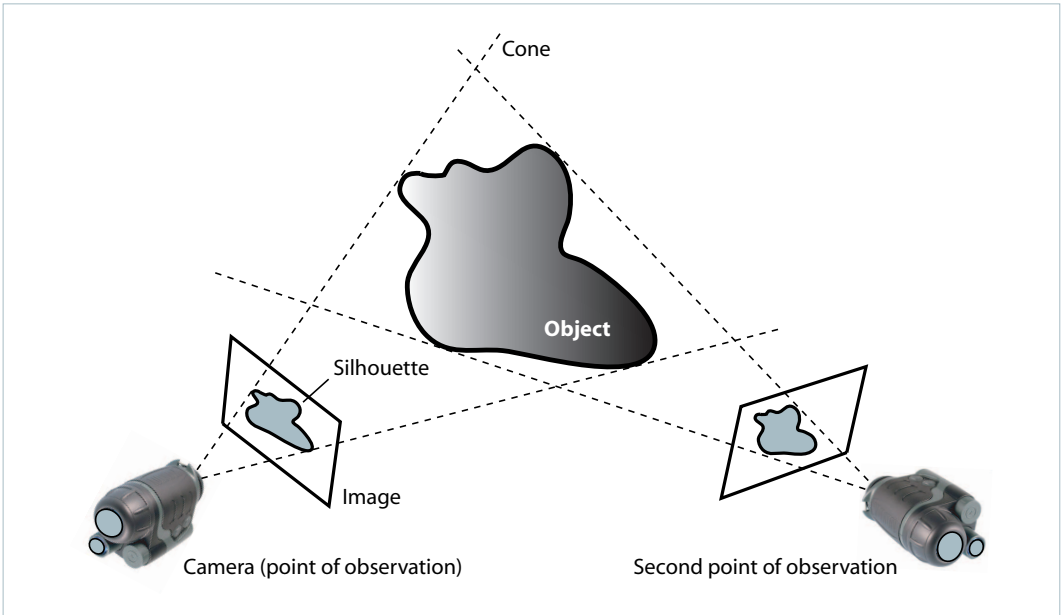


Figure 1.1: A simple approach to recover the shape of an object: shape from contours (or silhouettes). The silhouette and the point of observation for each view form a cone containing the object. The intersection of multiple cones is an estimate of object shape. Shape from contour techniques, however, fail at recovering object concavities.

far from having an ”ideal” vision system. Ultimately, one would like to be able to recover digital models quickly, accurately, at low cost, and, with minimal human intervention.

With that specific goal in mind, we developed a novel approach for 3D reconstruction of objects. We call this approach ”shadow carving”.

1.2 Shadow Carving: Combining Low Cost with Robust Shape Recovery

Shadow Carving is the first technique that incorporates ”shadow” information in a shape-from-contour reconstruction framework.

Shape from contours (or silhouettes), has been used for many years. An early shape from contours method was presented by Martin and Aggarwal [47] in 1983. The approach relies on the formation of a cone by a point of observation and the silhouette in an image obtained from that point, as shown in Fig. 1.1. All space outside of this cone must be outside of the object. Thus, the cone represents an upper bound conservative estimate of the object shape. By intersecting the cones formed from many different viewpoints, the estimate of object shape can be refined. Different techniques for computing the intersection of cones have

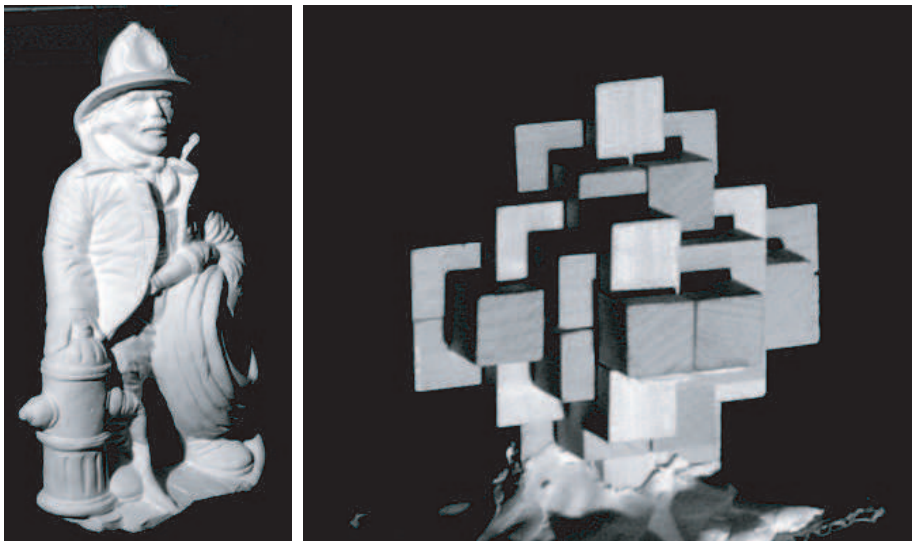


Figure 1.2: Objects concavities may be revealed by the presence of shadows that the object casts upon itself.

been proposed and subsequent research has improved on the efficiency of this approach. Because they are robust and conservative, techniques similar to Martin and Aggrawal's original method have found success in low-end commercial scanners. These approaches, however, are quite inaccurate when the object presents concavities, the reason being that concavities cannot be modelled using contours. As a result, severe artifacts are often generated in the final reconstruction of the model.

Computing shape from shadows – or shape from darkness – has also been studied for many years. Shafer and Kanade [63] established fundamental constraints that can be placed on the orientation of surfaces, based on the observation of the shadows one casts on another. Since then, several methods for estimating shape from shadows have been presented. All of these methods, however, rely on accurate detection of the beginning and ends of shadow regions. This is particularly problematic for attached shadows that are the end of a gradual transition of light to dark. Furthermore, existing shape from shadow methods essentially recover terrains with some hole estimates, rather than complete three dimensional objects.

A major advantage, however, is that shape from shadow methods allow the reconstruction of the shape in concavities. Such concavities would not appear in any silhouette of the object (see Fig. 1.2).

It is evident that to produce robust scanning systems it is useful to combine multiple shape-from-X approaches. A system is more robust if a shape estimated from shape-from-A is consistent with shape-from-B. Shadow carving uses this basic idea in combining shape from contour with shape from shadows. Shadow carving explores the information contained in the shadows that an object casts upon itself as it is lit from different directions. The starting point is an upper bound (conservative) estimate of the object's surface, i.e., a surface that completely contains the object. This upper bound approximation is estimated by measuring the

Figure 1.3: For specular objects, the brightness measured on the object's surface is *not* invariant with respect to the viewer's position. Rather, it depends on the position of both viewer and light source. Notice that the relative position between the highlight and the black cross changes as the viewer changes position.

object's contours. Then, we use the information provided by the object's shadows within an image. We make use of these cues to "carve" away pieces of the current surface estimate. We obtain a new estimate which is a closer approximation of the actual surface of the object. By adding images taken under different lighting, one may iterate this process. We proved that the algorithm generates a sequence of conservative estimates, and that the information provided by each new image is used efficiently.

Coupling "shadows" and "contours" information allow to combine robustness with low cost and good accuracy. The issue of low cost is worked out in that we propose a system that uses a commodity digital camera and controlled lighting systems composed of inexpensive lamps. Additionally, since our method relies on substantial variations in the intensities in acquired images, it avoids requiring the user to set parameters and hence, minimizes the user's intervention. Finally, since our technique progressively improves conservative estimates of surface shape, it avoids having an unstable method in which small errors accumulate and severely impact the final result.

We describe shadow carving in more detail in Chapter 2.

1.3 Beyond Matte Surfaces

Thus far, we have neglected another important limitation of traditional techniques for digital modelling. Most 3D reconstruction systems are based on the assumption that the brightness measured on the object's surface is invariant with respect to the viewer's position. Matte surfaces satisfy this assumption up to a certain degree; specular surfaces most often do not. Rather, the brightness measured on the surface depends on the position of both viewer and light source (see Fig. 1.3). Objects falling in this category are rather common (see Fig. 1.4), and represent a stumbling block for many current 3D reconstruction systems. Let us examine the reasons in more detail.

Algorithms based on feature-matching (stereo, structure from motion, voxel coloring) are based on the assumption that small surface features can be identified in images taken from different positions. Similarly, photometric stereo and shape from shading algorithms assume that the pixel brightness depends only on the position of the light source relative to the surface. Both hypotheses are not true for specular surfaces. In active lighting techniques, stripes or patterns of light are projected on the object surface from a laser or LCD projector. Unfortunately, shiny surfaces reflect most light along the specular direction, and therefore the signal observed by the imaging device might be either very low or measured in the wrong position. Techniques based on occluding contours require segmenting the object against the background. This task may become very hard as grazing-angle reflections of the surrounding scene make such boundaries 'melt' visually with the background. Shape-from-texture techniques exploit intrinsic features (albedo) of the diffuse components of the reflectance function. In order for these methods to work, it is necessary to rule out texture features due to reflections of the surrounding scene and only consider those due to the diffuse component.

Yet, the ability to recover the shape of highly specular surfaces is valuable in many applications such as:

- Industrial metrology: measurements of mechanical parts are needed for prototyping and quality control of industrial objects. Most objects of interest have highly reflective and polished surfaces (see Fig. 1.4, panel c).

- Diagnostic and control of space structures: large space structures are built of extremely light-weight components and thus, are highly susceptible to complex oscillations, as well to damage from debris. These surfaces are often metallic and highly reflective. On-line measurement of surface deformation is useful for stabilization and control. Diagnosis of impact damage is equally useful (see Fig. 1.4, panel e).

- Medical imaging: digital models need to be captured and displayed for diagnosis as well for on-line control of surgical procedures. Moist anatomical parts like the cornea of the eye and tissues that are visible endoscopically are highly reflective (see Fig. 1.4, panel f).

- Remote sensing of the surface of water: geophysical forces acting on water (i.e., wind, currents) may be studied through measurements of the shape of the surface (see Fig. 1.4, panel g).

- E-commerce: the possibility of viewing 3D interactive models of products over the web is appealing to customers. Many consumer products have highly glossy surfaces (see Fig. 1.4, panel a).

- Digital archival of art and heritage objects: artistic objects often contain highly specular surfaces. Jeff Koon's Rabbit (Fig. 1.4, panel d), is made of stainless steel and appears as an almost perfect mirror surface. Another example is the antique goblet in Fig. 1.4, panel b.

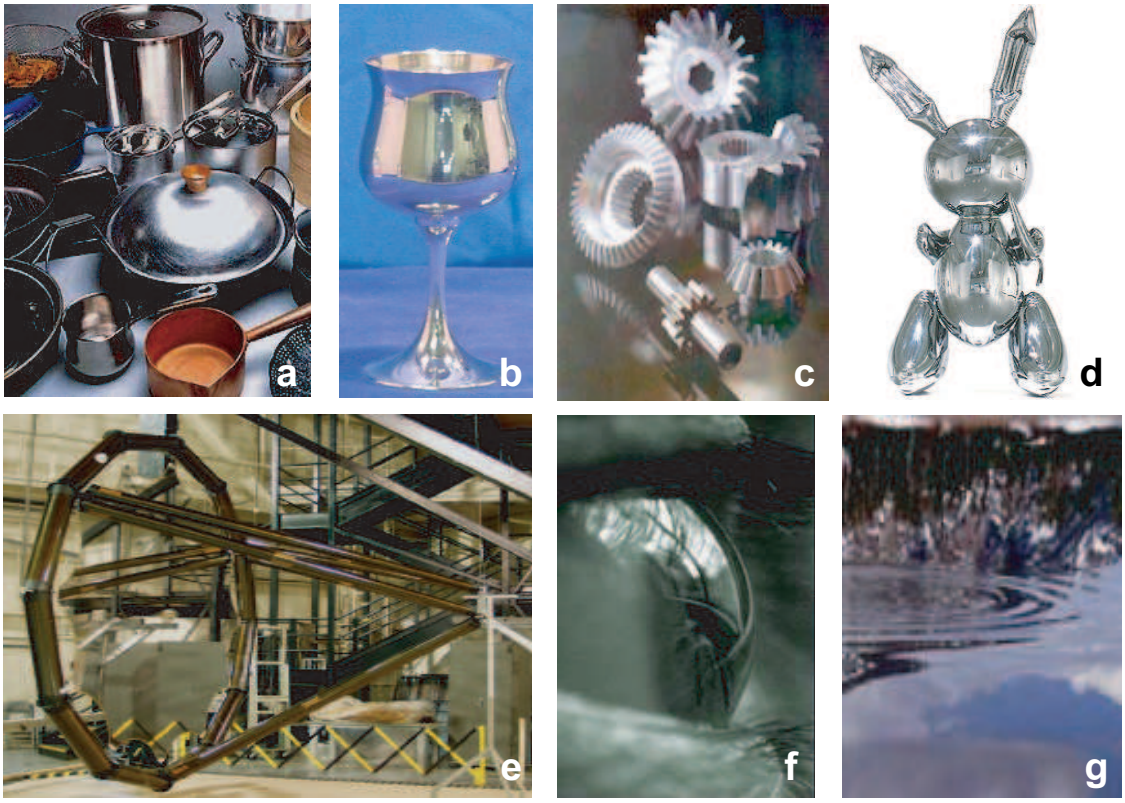


Figure 1.4: Applications. **Panel a**: E-commerce. **Panel b**: digital archival of heritage objects. **Panel c**: industrial metrology. **Panel d**: digital archival of art. Jeff Koons: Rabbit, 1986 stainless steel 41 x 19 x 12 inches -courtesy of The Broad Art Foundation. **Panel e**: diagnostic and control of space structures. Hexapod Structure - Tennessee State University / NASA Langley, courtesy of Dr. Gary A. Fleming. **Panel f**: medical imaging. **Panel g**: Remote sensing of the surface of water.

For shiny objects, one additional cue may be precisely the reflection of the environment. In fact, a curved mirrored surface produces distorted images of the surrounding world. For example, the image of a straight line reflected by a curved mirror is, in general, a curve. It is clear that such distortions are systematically related to the shape of the surface. Is it possible to invert this map and recover the shape of the mirror from the images it reflects? The problem is clearly highly under-constrained. By opportune manipulations of the surrounding world, we may produce almost any image from any curved mirror surface. This property is well known by artists, as illustrated by the anamorphic images that were popular during the Renaissance (see Fig. 1.5).

Researchers have made the problem of shape from reflections tractable by making assumptions about the position and shape of the object, and the structure of the surrounding world. Some authors focused on highlights (i.e., bright reflections of light seen on shiny surfaces such as the ones in Fig. 1.3) and the information they carry about the geometry of the surface. These studies revealed that even with a known

Figure 1.5: The problem of recovering the shape of a mirror from its specular reflections is highly under-constrained. By opportune manipulations of the surrounding world we may produce almost any image from any curved mirror surface. Examples are anamorphic images as shown in this figure. Anamorphic images are drawn by knowing the relationship between the proportions of a real object to the proportions of its reflected image on a mirror surface of known curvature. Once an anamorphic image is drawn, it can be restored to its corrected proportions by reflecting it in the mirror surface. Typical surfaces used for anamorphic images are cones and cylinders.

light source, specular highlights do not contain enough information to recover the shape univocally. Thus, the relevance and the use of additional geometrical constraints (multiple cameras and moving observer) were explored [36, 4, 84, 49, 82] . Often these efforts led to techniques requiring dedicated and expensive hardware equipment.

Other researchers tried to exploit the way in which a known surrounding scene is distorted when reflected off a mirror surface. Several techniques in the spirit of classic shape from structured lightning methods were proposed. Pioneering work by Sanderson, Nayar et al. [33] utilized an array of lights and the corresponding distribution of reflections. Among others, particularly interesting is the method proposed by Halsead [25] et al. where the shape of a specular surface is recovered by projecting circular pattern of lights. They applied their algorithm to the reconstruction of the cornea of the human eye: a very important problem in biomedicine. All these techniques, however, assume that a different degree of knowledge on shape and position of the object is available. Additionally, likewise common structured lighting techniques, and multiple images under

Figure 1.6: The task of recovering the shape of a mirror from its specular reflections may be very hard for the human visual system. The figure in the left panel shows a specular surface with a concavity in the center. It is quite difficult to perceive the actual shape of the concavity. Notice that object contours give very little information about the shape of the concavity. The figure in the right panel, shows the same specular surface seen from a different side. The geometry of the concavity becomes now much more clear. In this case, contours seem to play an important role in the perception of the shape, so we are interested in addressing the question: to what extent can the human visual system use specular reflections to perceive the shape of specular surfaces when other visual cues (such as shading, texture, contours) are absent?

different condition of the illuminant are required.

We propose a novel approach where the local shape of an unknown smooth specular surface can be recovered by observing its deforming behavior acting on the reflection of the scene. Our assumptions are that: i) the mirror surface is smooth; and ii) the reflected scene has known structure and position. Under these assumptions, we first describe a geometrical and algebraic characterization of how a patch of the scene is mapped into an image by a mirror surface. Then, we investigate the inverse problem and give necessary and sufficient conditions for recovering the local shape of the surface. Specifically, we prove that if the environment is composed of a planar patch of points or intersecting lines, then at any point the local shape can be recovered up to the third derivative.

Our technique differs and improves previous work in that it allows local shape reconstruction from a single image; it does not require prior assumption on object position or shape; it necessitates minimal and inexpensive hardware (one camera and a calibrated scene pattern only); local shape can be recovered in closed for solution; and the analysis is local and differential rather than global and algebraic.

The technique has been successfully implemented and several experiments measuring real specular surfaces have shown that the reconstruction is accurate. More details are presented in Chapter 3.

1.4 Can We See the Shape of a Mirror?

One important consequence of our computational analysis is that necessary and sufficient conditions for recovering the local shape of mirror surfaces are provided. In other words, the local shape of a mirror surface reflecting a scene can be recovered only if a certain amount of knowledge about that scene is available. Our computational study tells us precisely what this ”amount” is.

What happens, then, if the surrounding world is unknown? The problem is under-constrained and many solutions are possible. This observation leads naturally to the question: how do humans cope with the problem of perception of the shape of specular surfaces, given that most of time, we do not have the exact geometrical information about the world? An answer to this question would be of great value. The study of human vision can help us understand what cues may be used and which algorithms may be feasible, and ultimately guide our analytical efforts in the most productive direction.

While the perception of shape from a number of other visual cues (shading, texture, motion, stereoscopic disparity) has been extensively discussed in the human vision literature, surprisingly little is known to what extent the reflection of the environment off a specular surface may carry useful information for shape perception. The available knowledge is limited to the work by Beck, Koenderink and other researchers [36, 1, 73, 48]. Blake [5] showed that the human visual system successfully estimates shape and quality of a shiny object when the highlights are viewed stereoscopically. Recently, Fleming and collaborators [21] found in their experiments that the shape of mirror surfaces is readily perceived by human subjects. They explain their findings by noticing that specular reflections exhibit a different pattern of compression than surface texture. This feature would allow the human visual system to discriminate between these two cases and the pattern of compression would represent a cue for shape from specularities.

Through Chapter 4 of this thesis, we study this problem psychophysically and address the following questions: i) to what extent the human visual system can use specular reflections in isolation, namely, when other visual cues (such as shading, texture, motion, stereoscopic disparity) are absents ; ii) what are the underlying computational strategies for this task?

Our experiments surprisingly show that our subjects are very poor at judging the shape of mirror surfaces in absence of other visual cues. Figure 1.6 shows an example of how hard this task can be. We found, however, that for a considerable subset of the stimuli, subjects are highly consistent in their (most often wrong) perception. This observation leads us to the hypothesis that our subjects, rather than ‘computing’ a percept from each image based on geometrical considerations, may be associating a shape to each pattern in a stereotypical way, akin to pattern-matching. As mentioned earlier, this behavior is reasonable since the information available from specular reflections is ambiguous when the surrounding world is (partially)

unknown.

Chapter 2

Shadow Carving

2.1 Introduction

A number of cues, such as stereoscopic disparity, texture gradient, contours, shading and shadows, have been shown to carry valuable information on surface shape, and have been used in several methods for 3D reconstruction of objects and scenes. Techniques based on shadows have the advantage that they do not rely on correspondences, on a model of the surface reflectance characteristics, and they may be implemented using inexpensive lighting and/or imaging equipment. Past methods to recover shape from shadows, however, have proven to be sensitive to errors in estimating shadow boundaries. Additionally, their are mostly restricted to objects having a bas-relief structure.

We propose a novel 3D reconstruction method for using shadows that overcomes previous limitations. We assume that we have, as a starting point, a conservative estimate of object shape; that is, the volume enclosed by the surface estimate completely contains the physical object. We analyze images of the object illuminated with known light sources taken from known camera locations. We assume that we are able to obtain a conservative estimation of the object's shadows - that is, we are able to identify image areas that we are certain to be in shadow region, and do not attempt to find exact shadow boundaries. We do not make assumptions about the surface topology (multi-part objects and occlusions are allowed), although any tangent plane discontinuities over the objects surface are supposed to be detected. Our basic idea is that we compare observed shadows to those expected as if the conservative estimate were correct and adjust the current shape to resolve contradictions between the captured images and the current shape estimate. In this process, we incrementally remove (carve out) in a conservative way volume from the current object estimate in order to reduce the inconsistencies. Thus, a closer objects shape estimate is computed at each step. We call this novel procedure *shadow carving*. We provide a proof that the carving process is always conservative.

Shadow carving improves previous results on shape from shadows in two main aspects: i) it is more

robust with respect to the classification of shadow regions; ii) it gives the possibility of recovering objects in the round (rather than just bas-reliefs).

Our motivation for pursuing this work relies in applications where the user often has a very limited budget, and is primarily concerned with visually, rather than metrically, accurate representations. Furthermore, because users are often not technically-trained, the scanning and modeling systems must be robust and require minimal user intervention.

We validate our theoretical results by implementing a scanning system based on shape from silhouettes and shape from shadows. First, the silhouettes are used to recover an conservative estimate of the object shape. Then, a series of images of the object lit by an array of light sources are recorded with a setup shown schematically in Fig. 2.1. Such images are examined and the shadows that the object casts on itself are detected. The current shape estimate is then refined by the shadow carving procedure. Eventually, the improved estimate of shape can be further refined by methods that work well locally, such as photometric stereo [29].

Our system is designed to be inexpensive as other recently proposed schemes [53, 9]. It uses a commodity digital camera and controlled lighting systems composed of inexpensive lamps. Additionally, since our method relies on substantial variations in the intensities in acquired images, it does not requires precise tuning, hence it minimizes the user intervention. Finally, since our technique progressively improves conservative estimates of surface shape, it prevents small errors from accumulating and severely deteriorating the final results.

2.1.1 Chapter Organization

The rest of this chapter is organized as follows: we begin by reviewing previous work on shape from shadows and identifying the strengths and weaknesses of these methods in Section 2.2. We present the geometry of our approach and demonstrate that it always produces a conservative estimate of the object shape in Section 2.3. We propose our prototype system for shape recovery from silhouettes and shadows in Section 2.4. We test the performance of the method with different configurations of lights and camera positions and assess the accuracy of the reconstruction due to error in the shadow estimates in Section 2.6.1. Finally, we present results we have obtained from a small table-top implementation of our system in Section 2.6.2.

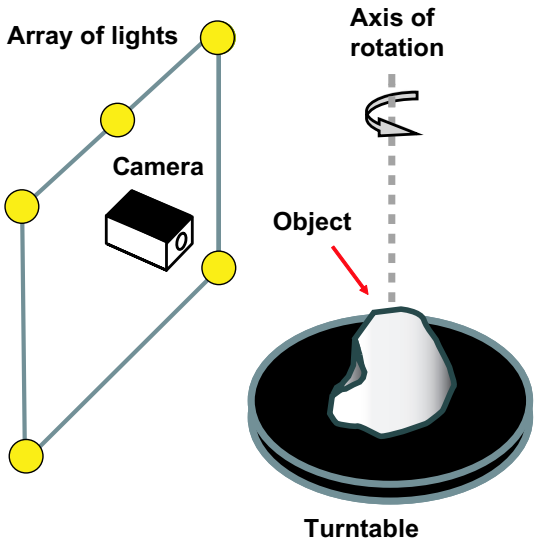


Figure 2.1: Setup for shape from self-shadowing: Using each of several lights in turn, and a camera in front, allows multiple sets of shadow data to be obtained for each object position.

2.2 Background

Computing shape from shadows – or shape from darkness – has a long history. Shafer and Kanade [63] established fundamental constraints that can be placed on the orientation of surfaces, based on the observation of the shadows one casts on another. Hambrick et al. [26] developed a method for labelling shadow boundaries that enables inferences about object shape. Since then, several methods for estimating shape from shadows have been presented. Because we aim at designing a 3D scanner, we focus on reconstruction schemes where the light source position is known, rather than the case of unknown light source direction (e.g., [37]). Also, we limit our analysis to methods using self-shadows (i.e., shadows cast by the object upon itself) rather than shadows cast by external devices as in [9]. Hatzitheodorou and Kender [27] presented a method for computing a surface contour formed by a slice through an object illuminated by a directional light source casting sharp shadows. Assuming that the contour is defined by a smooth function – and that the beginning and end of each shadow region can be found reliably – each pair of points bounding a shadow region yields an estimate of the contour slope at the start of the shadow region, and the difference in height between the two points as shown in Fig. 2.2. The information for shadows from multiple light source positions is used to obtain an interpolating spline that is consistent with all the observed data points.

Raviv et al. [51] developed an extended shape from shadows method. The object is set on a known reference surface with a camera directly above. A series of images is captured as a collimated light source moves in an arc over the surface. For the 2D reference surface, a volume of data is then obtained with

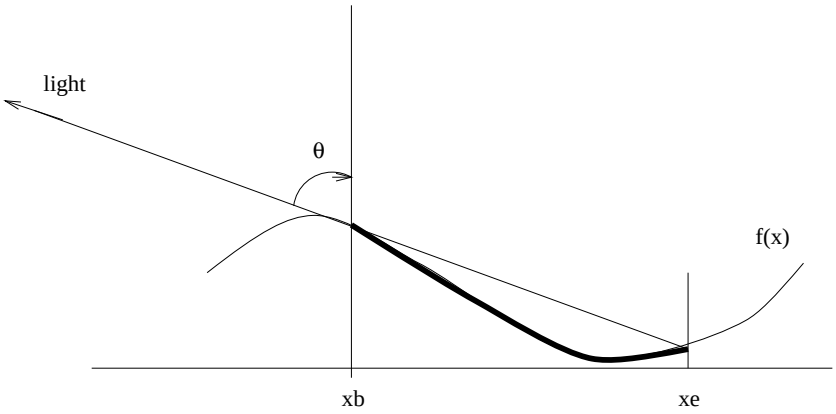


Figure 2.2: Shape from Shadows: For terrain surfaces $f(x)$ and a known light source direction θ, $f'(x_b) = tan\theta$, and $f(x_b) - f(x_e) = f'(x_b)(x_e - x_b)$. Using data for many angles θ an estimate of the continuous function $f(x)$ can be made.

the third coordinate being the angle of the light source to the reference surface, and the volume recording whether the reference surface was in shadow for that angle. A slice through this volume is referred to as a *shadowgram*. Similar to Hatzitheodorou and Kender, by identifying beginning and ending shadow points for each light position, the height difference between the points can be computed. Also, by observing the change in shadow location for two light source positions, the height between the start of the shadow at one position relative to the other can be found by integration. As long as shadow beginnings and endings can reliably be detected, the top surface of the object can be recovered as a height field. Furthermore, by detecting splits in the shadowgram (i.e., positions that have more than one change from shadowed to unshadowed), holes in the surface below the top surface can be partially recovered.

Langer et al. [39] extend the method of Raviv et al. for computing holes beneath the recovered height field description of the top surface for two dimensions. They begin with the recovered height field, an NxN discretization of the two dimensional space, and the captured shadowgram. Cells in this discretization are occupied if they are in the current surface description. Their algorithm steps through the cells and updates them to unoccupied if a light ray would have to pass through the cell to produce a lit area in the captured shadowgram.

Daum and Dudek [16] subsequently developed a method for recovering the surface for light trajectories that are not a single arc. The estimated height field description is in the form of an upper bound and lower bound on the depth of each point. The upper and lower bounds are progressively updated from the information obtained from each light source position.

All of these methods rely on accurate detection of the boundaries of shadow regions. This is particularly problematic for attached shadows that are the end of a gradual transition of light to dark. Height estimates

that use gradients derived from the estimate of the start of attached shadows are particularly prone to error. Yang [79] considers the problem of shape from shadows with error. He presents a modified form of Hatzitheodorou and Kender approach, in which linear programming is used to eliminate inconsistencies in the shadow data used to estimate the surface. While the consistency check does not guarantee any bounds on the surface estimate, it does guarantee that the method will converge. He shows that the check for inconsistences is NP-complete. While more robust than Hatzitheodorou and Kender's method when applied to imperfect data, Yang's technique is still restricted to 2.5D terrains. Finally, Yu and Chang [80] give a new graph-based representation for shadow constraints.

Our method does not require a restriction to 2.5D terrains. Rather, it allows a fully 3D reconstruction of the object. Additionally, we do not rely on knowing the boundaries of shadow regions to compute surface shape. Similar to Yangs approach, we use the idea of consistency to avoid misinterpreting data. However, rather than comparing multiple shadow regions for consistency, we check that observed shadow regions are consistent with our current surface estimate.

Our proposed approach - shadow carving - is similar in spirit to the space carving approach of Kutulakos and Seitz [38]. Our approach differs from [38] in that we consider consistency between a camera and light views, rather than multiple camera views. Consistency can be tested robustly by detecting shadows, without requiring the hypothesis of Lambertian surface. We begin with a conservative surface definition, rather than a discretized volume. Inconsistent regions can be carved out by moving surface points at the resolution of the captured images, rather than being limited to a set of fixed resolution voxels. Most importantly, we provide a proof of correctness that well defined portions of volume can be removed in a conservative manner from the current object estimate, instead of iteratively removing single voxels until all the inconsistencies are solved.

This chapter gathers our own previous work [61, 60] and presents an extended experimental analysis in that: i) performance of the method with different configurations of lights and camera positions is tested; ii) accuracy of the reconstruction due to errors in the shadow estimates is assessed; iii) throughout experiments with real objects are shown.

2.3 Shadow Carving

We introduce a formal definition of shadow carving and present its main property of yielding conservative object estimates in section 2.3.1. Then we show a simple example of shadow carving in the context of the epipolar slice model described in sections 2.3.2 and 2.3.3. Finally, we prove that shadow carving always yields conservative estimates in sections 2.3.4, 2.3.5, 2.3.6 and 2.3.7.

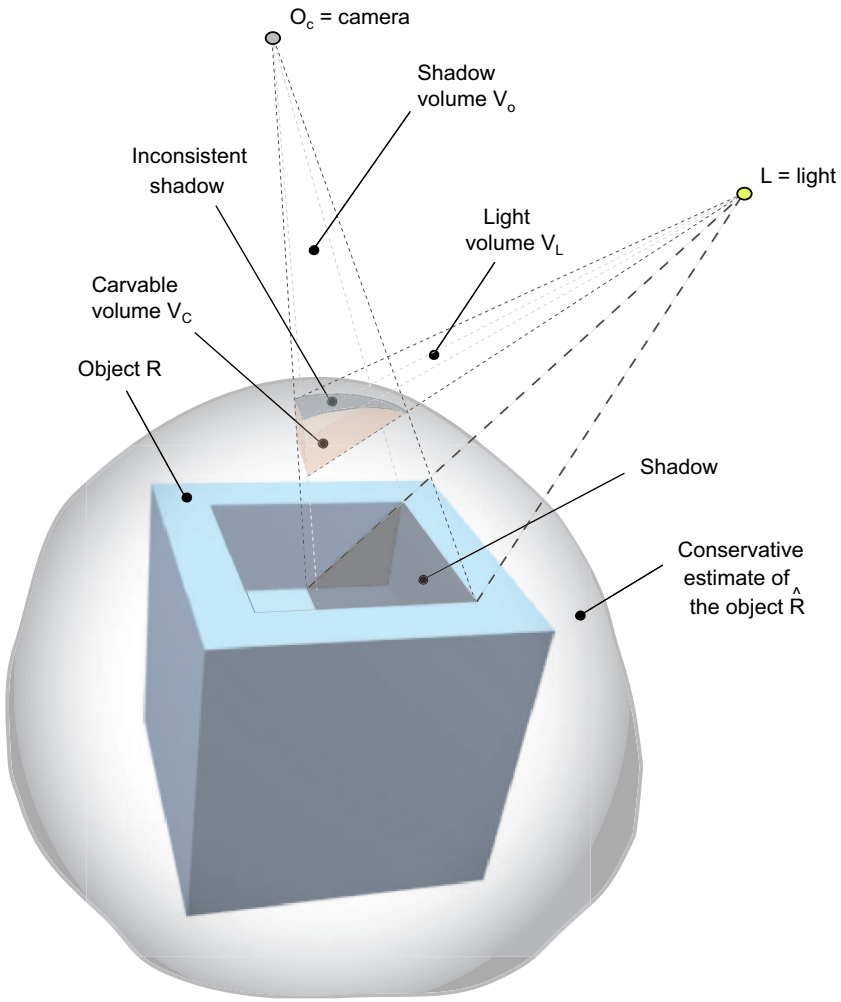


Figure 2.3: Example of shadow carving in 3D.

2.3.1 The Shadow Carving Theorem

Consider an object in 3-D space and a point light source L illuminating it. See Fig. 2.3. One or more shadows are cast over the object by parts of the object itself. The scene is observed by a pin-hole camera whose center is located in O_c. Let us introduce the following definitions:

Definition 2.3.1 *A* conservative object estimate *is any volume $\hat{R}$ that contains the object R.*

Definition 2.3.2 *The* shadow volume *V_o is the set of lines through the center of the camera projection O_c and all of the visible points of the object's surface that are in shadow.*

Definition 2.3.3 *The* inconsistent shadow *is the portion of the surface of the conservative object estimate that intersects V_o and is visible from the camera and the light source.*

Definition 2.3.4 *The* light volume V_L *is the set of lines through the light source* L *and the points of the inconsistent shadow. We denote the dependency of* V_L *on the particular choice of* $\hat{R}$ *by* $V_L(\hat{R})$.

Definition 2.3.5 *The* carvable volume $V_C(\hat{R})$ *is* $V_o \cap V_L(\hat{R}) \cap \hat{R}$*, where* $\hat{R}$ *is a conservative object estimate.*

Theorem 2.3.1 *If the object surface is smooth,* $\hat{R}$ *minus* $V_C(\hat{R})$ *is a conservative object estimate.*

Notice that all the quantities (i.e., L, O_c, $\hat{R}$ and the image points of the object's surface that are in shadow) are available either from calibration or from measurements in the image plane. Therefore theorem 2.3.1 suggests a procedure to estimate the object incrementally: i) start from a conservative object estimate; ii) measure in the image plane all of the visible points of the object's surface that are in shadow and compute the shadow volume; iii) compute the corresponding inconsistent shadows; iv) compute the corresponding light volume; v) intersect the conservative object estimate, the shadow volume and the light volume and calculate the carvable volume; vi) remove the carvable volume from the conservative object estimate to obtain an new object estimate. Theorem 2.3.1 guarantees that the new object estimate is still a *conservative* object estimate. The process can be iterated by considering a new light source or by viewing the object from a different vantage point. This procedure will be developed in details in section 2.4.

As we shall see in section 2.3.4, the theorem still holds if there are errors in detecting the visible points of the object's surface that are in shadow. These errors, however, must be conservative; namely, a piece of shadow can mislabeled as non-shadow, but a non-shadow cannot be mislabeled as a shadow.

2.3.2 The Epipolar Slice Model

In order to prove Theorem 2.3.1 we examine the problem in an appropriately chosen 2-D slice of the 3-D space, *the epipolar slice*. As we shall discuss in more details in section 2.3.6, the results that we prove for a given slice hold in general and do not depend on the specific choice of the slice. Thus, the extension to the 3-D case is immediate by observing that the epipolar slice can sweep the entire object's volume.

Consider the family of planes Π_L passing through O_c and L (see Fig. 2.4). Each plane $\pi_L \in \Pi_L$, intersects the image plane π_i and the object. In other words, each π_L defines an epipolar slice of 3-D space. For each slice, we have the *image line* (i.e., the intersection of π_i with π_L), the *image shadow segment* (i.e., the intersection of the estimated image shadow with π_L), the *object contour* P (i.e., the intersection of the object's surface with π_L) and the *conservative object contour* $\hat{P}$ (i.e., the intersection of the conservative surface estimation with π_L). We can also have the *object area* A_P (that is, the area bound by P) and the *conservative object area* $A_{\hat{P}}$ (that is, the area bound by $\hat{P}$).

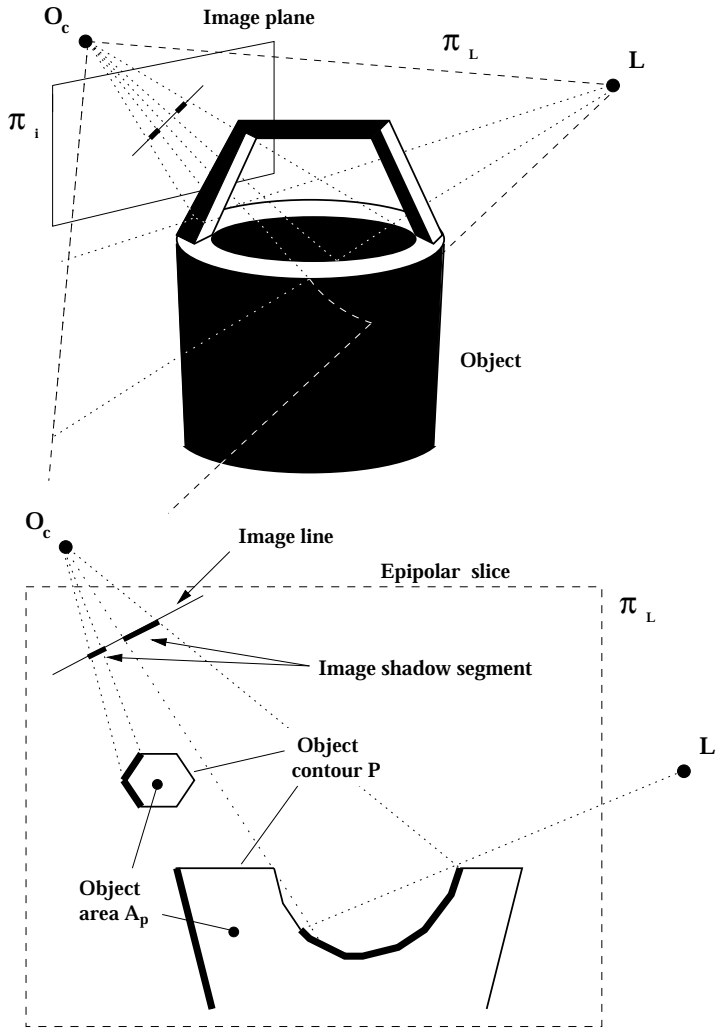


Figure 2.4: Top: An object in 3-D space and a point light source L illuminating it. The scene is observed by a camera whose center is located in O_c and whose image plane is called π_i. Bottom: The plane π_L defines an epipolar slice of 3-D space.

2.3.3 Example

Fig. 2.5 shows an example of shadow carving in the epipolar slice model. The shadow $\bar{s}$ is cast by the the object contour P over itself. s is the image of $\bar{s}$. The *shadow area* A_o is the set of lines through O_c and all of the point along s. The portion of $\hat{P}$ that is visible from the camera and the light, and intersects A_o is the *inconsistent shadow* $\hat{s}$. This shadow is called inconsistent because it is visible from the light source L. Thus, the conservative estimate $\hat{P}$ has to be re-adjusted in order to explain this inconsistency. The *light area* A_L is the set of lines through L and all of the points on $\hat{s}$ that are visible from L. Finally, $A_o \cap A_L \cap A_{\hat{P}}$ gives the area A_C which we call the *carvable area* (i.e., cross-hatched area in the figure). A_C can be removed from $\hat{P}$, generating an improved conservative approximation of P. Notice that the new conservative estimate

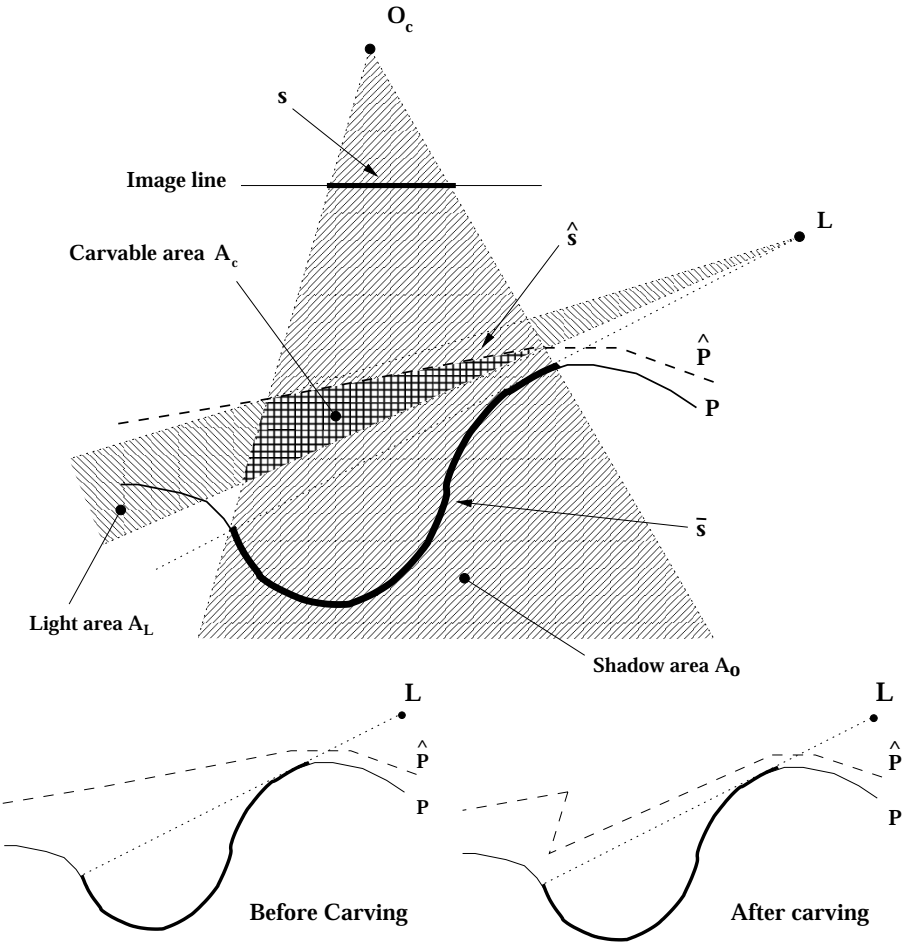


Figure 2.5: Example of shadow carving.

is consistent with the observed shadow s and with the light source position. Thus, no additional carving is required. Finally, notice that A_C can be removed in one shot rather than by means of an iterative process as in [38].

Many questions arise and we may wonder what happens if i) the topology of P is more complex; ii) the shadow is occluded by other object parts; iii) multiple shadows are imaged by the camera; iv) the object surface contains low-albedo regions that do not allow a correct or complete estimate of the shadows. Can we still define a carvable area? Can we guarantee that the carvable area is always outside the object (i.e., the new estimate is still conservative)?

In Section 2.3.4, we address these questions; we first introduce the definition of atomic shadow and then describe how the measured shadow can be decomposed into atomic components. Given such decomposition, we formally define the carvable area A_C and prove that A_C is always outside the object area (sections 2.3.5 and 2.3.6). As we shall see, no hypothesis on the topology of object is made. However, an hypothesis on the

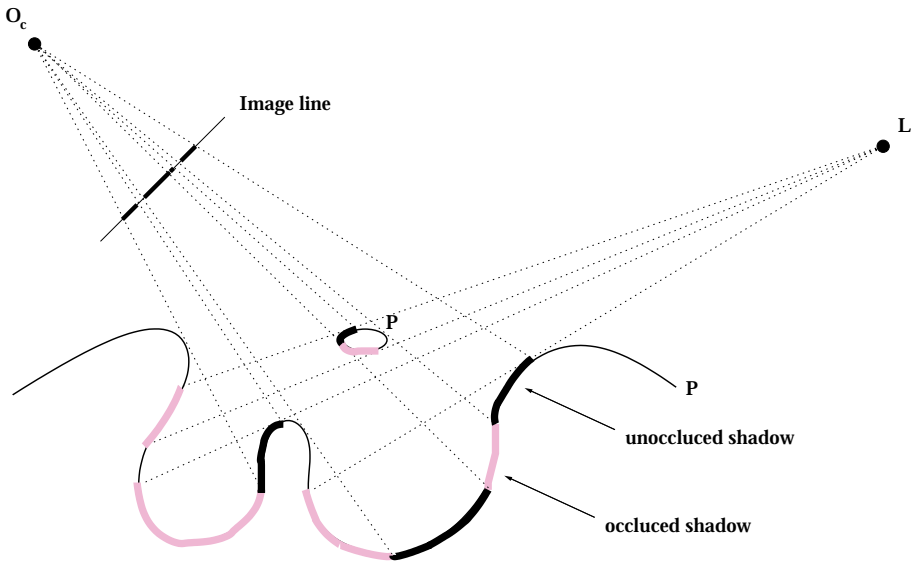


Figure 2.6: Example of occluded and unoccluded shadows.

smoothness of the object surface is required and is discussed at the end of Section 2.3.4.

2.3.4 The Shadow Decomposition

Let us consider an epipolar slice of a 3-D object. In general, the object's contour P on the epipolar slice might comprise more than one separate contour components. For instance, the body of the bucket in Fig. 2.4 produces the contour P_1 and the handle produces the contour P_2.

Given a point light source L, some portions of the object contour will be visible from the light source, whereas other portions will not.

Definition 2.3.6 *A* shadow *is a portion of contour that is not visible from the light source.*

Thus, depending on the object's topology and the light source position, there will be a certain shadow distribution.

In general, some portions of a given shadow might be occluded from the camera view by other parts of the object. A portion of shadow which is visible from the camera (i.e from the center of projection) is called *unoccluded*. Fig. 2.6 shows examples of unoccluded shadows: the unoccluded shadows are indicated with the bold black lines; the occluded portions of shadow are indicated with bold pink lines. It is clear that whether a shadow is occluded or not only depends upon the contour topology and the camera position.

Let us suppose that we have a technique to detect shadows. It is a reasonable assumption that such a shadow detection technique is always *conservative*. That is, a shadow may not be detected, but whatever

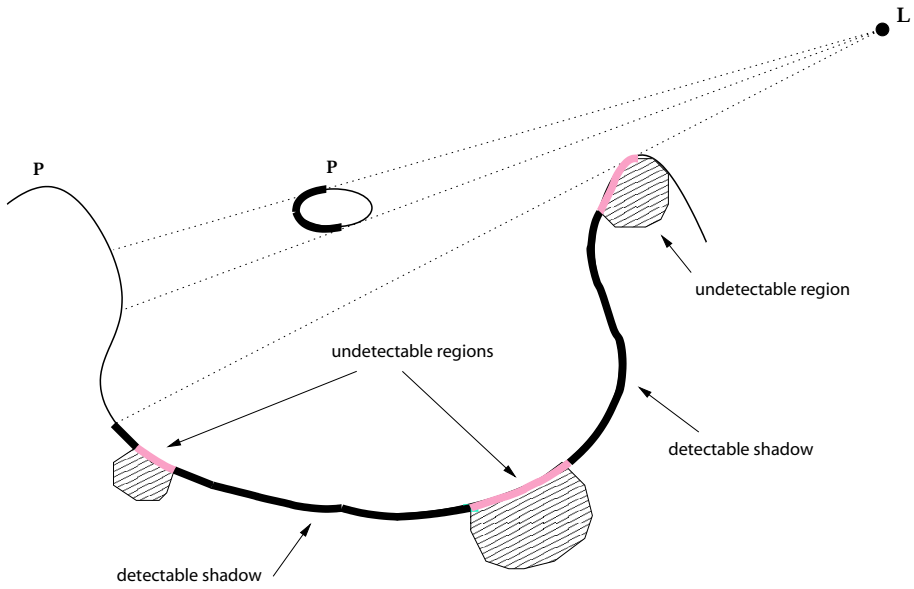


Figure 2.7: Example of detectable shadows.

is labeled as shadow is indeed a shadow. See section 2.4.4 for details. Thus, a portion of contour is called *undetectable* if, according to the shadow detection technique, it cannot be classified either as a shadow or as not a shadow. We define *detectable* as a shadow which does not lie within an undetectable portion of contour. Fig. 2.7 shows examples of detectable shadows and undetectable regions: the hatched areas are undetectable contour regions; the detectable shadows are indicated in bold black.

Definition 2.3.7 *A maximal connected portion of shadow which is both unoccluded and detectable is called* atomic.

An atomic shadow is indicated with the symbol $\bar{a}$ and its corresponding perspective projection into the image line is indicated by a. We call a an atomic image shadow. As a result, any shadow $\bar{s}_j$ can be decomposed into its atomic components $\bar{a}_{j,1}$, $\bar{a}_{j,2}$ ... $\bar{a}_{j,k}$. See Fig. 2.8 for an example: the atomic shadows (indicated in bold black) within $\bar{s}_1$ are $\bar{a}_{1,1}$, $\bar{a}_{1,2}$, $\bar{a}_{1,3}$ and $\bar{a}_{1,4}$. The perspective projection of the atomic shadows into the image line yields the atomic image shadows $a_{1,1}$, $a_{1,2}$, $a_{1,3}$ and $a_{1,4}$.

We assume that the shadow detection technique gives us an estimation e_u of the perspective projection into the image line of the complete set of unoccluded shadows and an estimation e_d of the perspective projection of the complete set of detectable regions. The intersection of e_u with e_d gives the estimated shadow distribution e along the image plane of both unoccluded and detectable shadows. The estimated shadow e can be described as the union of maximal connected components which we call estimated shadow component e_i. It is easy to show that there is no unique correspondence between estimated shadow components and atomic image

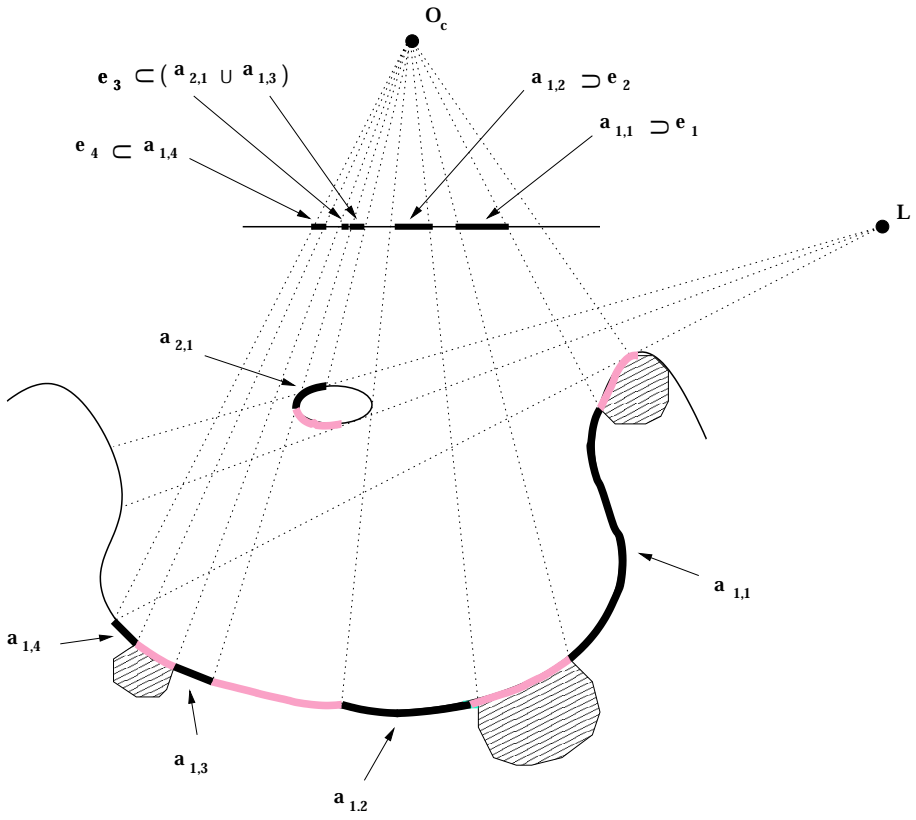


Figure 2.8: Example of atomic shadows.

shadows. For instance, see Fig. 2.8. The estimated shadow component e_3 corresponds to atomic image shadows $a_{2,1}$ and $a_{1,3}$.

Lemma 2.3.1 *If the object contour's first order derivative is continuous (i.e., the object's contour is smooth) and if the shadow detection is conservative (i.e., a piece of shadow can be mislabeled as non-shadow, but a non-shadow cannot be mislabeled as a shadow), an estimated shadow component is always a lower bound estimation of either an atomic image shadow or the union of two or more atomic image shadows.*

Proof The lemma just follows from the definitions. □

An example is shown in Fig. 2.8: the estimated shadow e within the image line can be decomposed into its estimated shadow components e_1, e_2, e_3 and e_4. In particular, e_1 is a lower bound estimation of $a_{1,1}$. e_3 is a lower bound estimation of the union of $a_{1,3}$ and $a_{2,1}$. Notice that e_3 appears as a connected shadow component, although $a_{1,3}$ and $a_{2,1}$ are the images of atomic shadows generated over two different contours.

In the following we want to show that if the hypothesis of smoothness is removed, Lemma 2.3.1 is no longer verified. Let us consider the example depicted in Fig. 2.9. The surface P casts two shadows over P itself. $\bar{a}_1$ and $\bar{a}_2$ are the corresponding atomic shadows. In particular, $\bar{a}_1$ is cast by the point $\bar{p}$. Hence, $\bar{p}$

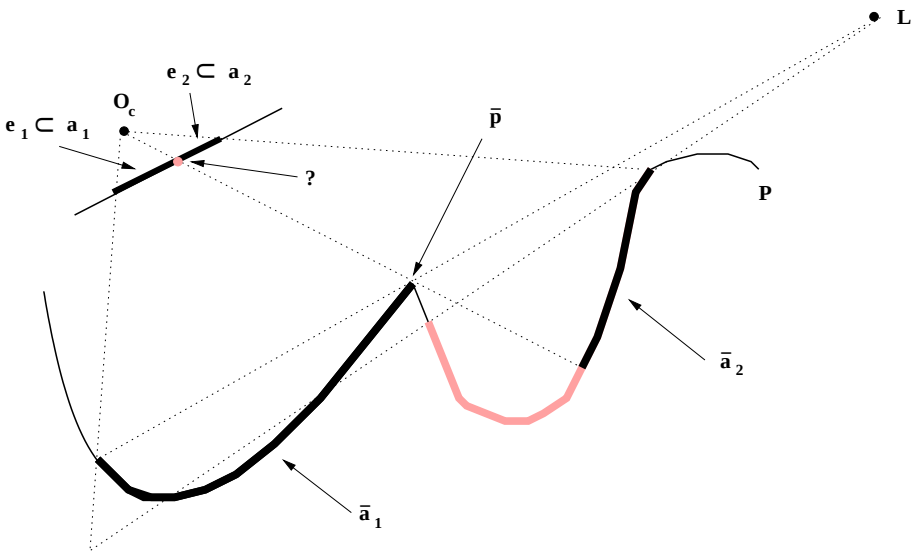


Figure 2.9: Contour with singular point.

cannot be in shadow. Furthermore, $\bar{p}$ is visible from O_c. Thus, the corresponding atomic image shadows a_1 and a_2 do not constitute a connected component. However, if the edge is sharp enough, the camera might not be able to resolve $\bar{p}$, which will be estimated as being in shadow. The shadow decomposition fails; the estimated shadow component is no longer a lower bound conservative estimate of a_1 and a_2. In other words, a_1 and a_2 are estimated to be a unique connected component instead of two disconnected shadow components e_1 and e_2. As a result, Lemma 2.3.1 does not necessarily hold when the contour smoothness hypothesis is not verified.

We can remove the hypothesis of smoothness if we suppose that we have available a technique to identify points whose first order derivative is not continuous. We call such points *singular* and we label them as undetectable. Let us consider again the example of Fig. 2.9. If $\bar{p}$ can be identified as singular and therefore its image p labeled as undetectable, a_1 and a_2 are no longer estimated as a unique shadow component, but as two disconnected estimated shadows components e_1 and e_2. The shadow decomposition is again conservative and Lemma 2.3.1 is verified. From now on, we assume either to deal with smooth contours or to have a technique to identify singular points so that we can use the property stated in Lemma 2.3.1.

In the next section we formally define the carvable area and prove that each carvable area attached to a generic e_i does not belong to the object area (conservative carving). By means of Lemma 2.3.1 we decompose the problem into two cases: an estimated shadow component corresponds to an atomic image shadow; an estimated shadow component corresponds to the union of two or more atomic image shadows.

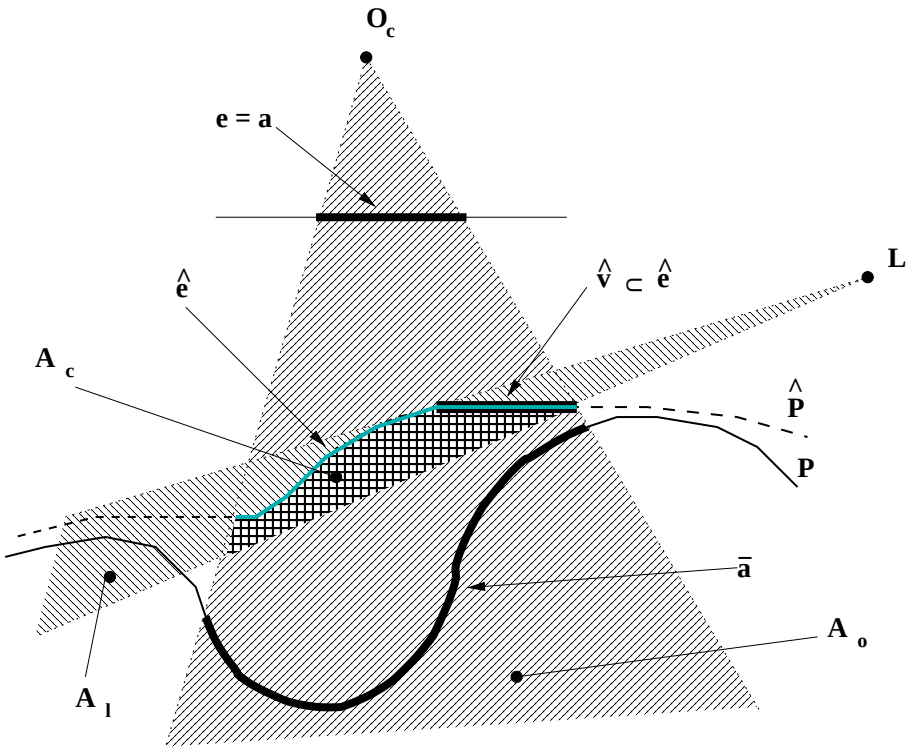


Figure 2.10: The atomic shadow case.

2.3.5 The Atomic Shadow Case

Let us consider an estimated shadow component e and let us assume that e is exactly the atomic image shadow a (see Fig. 2.10). We call $\bar{a}$ the corresponding atomic shadow over the object contour. Let A_o be the area defined by the set of lines through O_c and e. A_o is the 2D slice of the shadow volume V_o in definition 2.3.2. The following lemma holds:

Lemma 2.3.2 *Given an estimated shadow component e and the corresponding shadow area A_o, all the points belonging to P and within A_o either belong to $\bar{a}$ or are not visible from O_c. In particular, if there is a point $p \in P$ and $\in A_o$ not visible from O_c, there must exist at least one point $q \in \bar{a}$ between p and O_c.*

Proof. The lemma follows from the definition of atomic shadow and Lemma 2.3.1. □

In the following, we introduce the geometric quantities necessary to formally define the carvable area. Let $\hat{P}$ be the current conservative contour estimation and let $\hat{e}$ (light gray line in Fig. 2.10) be the projective transformation (centered in O_c) of e onto the conservative contour $\hat{P}$ closest to O_c[1] Since $\hat{P}$ is a conservative

[1]We define a segment $\hat{s}$ (e.g., a portion of the conservative estimate of object contour) to be *closer* than another segment $\bar{s}$ (e.g., a portion of object contour) with respect to a point Q, as follows. Let $\hat{p}$ be a point belonging to $\hat{s}$. If the line through Q and $\hat{p}$ intersect $\bar{s}$, the intersection defines a point $\bar{p}$ on $\bar{s}$. If $distance(Q, \hat{p}) \leq distance(Q, \bar{p})$ for any $\hat{p} \in \hat{s}$, then $\hat{s}$ is defined to be closer to Q then $\bar{s}$. Finally, we define a segment $\hat{s}$ to be the *closest to* Q if $\hat{s}$ is closer to Q than any other segment (or point) $\bar{s}$ in the plane.

estimation of the real P, $\hat{e}$ is closer to O_c then $\bar{a}$. Let $\hat{v}$ (black line in Fig. 2.10) be the portion of $\hat{e}$ whose points can be connected by a line segment to L without intersecting $\hat{P}$. Thus, any point $\in \hat{v}$ must be visible from L. $\hat{v}$ corresponds to the inconsistent shadow in definition 2.3.3. Let A_l be the area defined by the family of lines passing through L and any point along $\hat{v}$. A_l is the 2D slice of the volume V_L in definition 2.3.4.

Definition 2.3.8 *We define the* carvable area A_C *as the area obtained by the intersection of the estimated object area (bounded by* $\hat{P}$*),* A_l *and* A_o*.*

The carvable area corresponds to the cross-hatched area in Fig. 2.10 and is the 2D slice of the volume A_C in definition 2.3.5. The key result in this section is stated in the following lemma:

Lemma 2.3.3 *The carvable area* A_C *does not intersect the real object contour* P*.*

Proof. Suppose, by contradiction, that there is a portion q of the object contour P within A_C. There are two possible cases: q is fully included within the atomic shadow $\bar{a}$ (case 1); q is not fully included within $\bar{a}$ (case 2). Let us start with case 1. It is easy to conclude that, since A_C is defined by $\hat{v}$ (which is fully visible from L), there must exist at least one point $p \in q \subseteq \bar{a}$ visible from L. The contradiction follows from the definition of shadow 2.3.6. Consider case 2: let q_p be a portion of q not included in $\bar{a}$. Since $\bar{a}$ is atomic, by Lemma 2.3.2, q_p must eventually be occluded by a portion $\bar{a}_p$ of $\bar{a}$. But since $\hat{e}$ is closer to O_c than $\bar{a}$ and since q_p belongs to A_C, $\bar{a}_p$ must lie within A_C. Hence we can apply case 1 to $\bar{a}_p$ achieving again the contradiction. The lemma is fully proved. □

Lemma 2.3.4 The carvable area A_C cannot completely lie within the object.

Proof. The lemma holds because, by definition, $\hat{e}$ is closer to O_c then $\bar{a}$. □

Proposition 2.3.1 *Any point within a carvable area cannot belong to the actual object area.*

Proof. The proposition follows from Lemma 2.3.3 and 2.3.4. □

2.3.6 The Composite Shadow Case

The composite shadow case arises when e is not necessarily attached to a single atomic shadow (e.g., Fig. 2.11). Let us assume that e is actually composed by the union of J atomic image shadows $a_1, a_2, \ldots a_J$. The corresponding atomic shadows lying over the object contour are $\bar{a}_1, \bar{a}_2, \ldots \bar{a}_J$, respectively. The area A_o is defined by e as in the atomic case. Lemma 2.3.2 can be modified as follows:

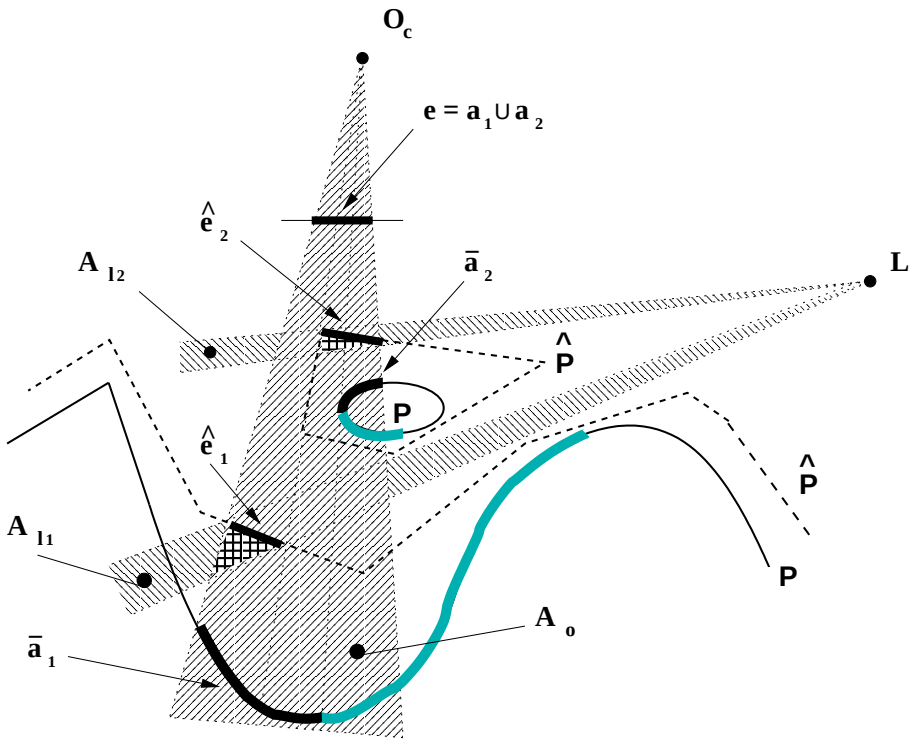


Figure 2.11: The composite shadow case.

Lemma 2.3.5 *Given an estimated shadow component e and the corresponding A_o, all the points belonging to P and within A_o either belong to one of the atomic shadows $\bar{a}_1, \bar{a}_2, \ldots \bar{a}_J$ or they are not visible from O_c. In particular, if there is a point $p \in P$ and $\in A_o$ not visible from O_c, there must exist at least one point $q \in \{\bar{a}_1, \bar{a}_2, \ldots \bar{a}_J\}$ between p and O_c.*

Proof. The lemma directly follows from the definitions of atomic shadows, Lemma 2.3.1 and the assumption that e is a connected shadow component. □

We can define $\hat{e}$ as in the atomic case. The difference here is that $\hat{e}$ may be decomposed into K different components $\hat{e}_1 \ldots \hat{e}_K$ located in different positions within A_o, depending on the topology of $\hat{P}$. For each k, we can define $\hat{v}_k$ as the portion of $\hat{e}_k$ whose points can be connected by a line segment to L without intersecting $\hat{P}$. Furthermore, for each $\hat{v}_k$ we define a corresponding A_{l_k} as in the atomic case.

Finally:

Definition 2.3.9 *We define the* set of carvable areas $A_{C_1}, A_{C_2}, \ldots A_{C_K}$ *attached to the estimated shadow component e as set of areas obtained by the intersection among the estimated object area (bounded by $\hat{P}$), A_o and the set $A_{l_1}, A_{l_2}, \ldots \ldots A_{l_K}$.*

Proposition 2.3.2 *Any point within the set of carvable areas $A_{C_1}, A_{C_2}, \ldots A_{C_K}$ cannot belong to the actual*

object area.

Proof. The proposition follows from Lemma 2.3.5 and by modifying accordingly Lemma 2.3.3, Lemma 2.3.4 and the corresponding quantities. □

Proposition 2.3.1 and Proposition 2.3.2 guarantee that points taken from any of the carvable areas, are always outside the real object area. The extension to the 3-D case is immediate by observing that: i) the epipolar slice can sweep the entire object's volume; ii) the interplay of light and geometry that allows us to prove Proposition 2.3.1 and Proposition 2.3.2 take place within each slice and is independent of the other slices. Thus, Theorem 2.3.1 is fully proven.

Notice that Theorem 2.3.1 holds under the hypothesis of Lemma 2.3.1: i) the actual object's contour is smooth (or a technique is available to identify points whose first order derivative is not continuous - see section 2.3.4); ii) the shadow detection technique is *conservative* - that is, a shadow may not be detected but whatever is labeled as shadow is indeed a shadow.

2.3.7 Effect of Errors in the Shadow Estimate

What happens when errors in the shadow estimation occur? We proved Proposition 2.3.1 with the assumption that an estimated shadow component is equivalent to an atomic image shadow (or the union of two or more atomic image shadows). In the presence of a conservative error in the shadow estimation, the estimated shadow component is always included in the atomic image shadow. Thus, the corresponding carvable areas will be a subset of those obtained from the full atomic shadow. In the limit case where no shadows are detected, no volume is removed from the object. As a result, we still perform a conservative carving. This property makes our approach robust with respect to conservative errors in identifying shadows. If a non-conservative error in the shadow estimation occurs, however, the estimated shadow components may not necessarily be included in the corresponding atomic image shadows. As a result, some carvable areas may be contained within the actual object's boundaries. The shadow carving procedure is no longer conservative. In section 2.6, we will show an example of non conservative carving due to non conservative errors in the shadow estimate.

As a conclusion, an estimated carvable area is always guaranteed to be outside the actual object boundaries as long as errors in the shadow estimation are conservative and singular points along the object contour are detectable.

2.4 A System for 3D Reconstruction from Silhouettes and Shadows.

In this section, we present the design of a system for recovering shape from shadow carving. The system combines techniques of shape from silhouettes and shadow carving. We first briefly review techniques based on shape from silhouettes in Section 2.4.1. Then we present our implementation of shape from silhouettes. We describe a new hardware configuration that tackle the difficult problem of extracting the object image silhouette in Section 2.4.2. The outcome of shape from silhouettes is an initial conservative estimate of the object. We review hybrid approaches in Section 2.4.3 and show that the initial estimate can be refined by shadow carving in Section 2.4.4.

2.4.1 Background: Shape from Silhouettes

The approach of shape from silhouettes − or shape from contours − has been used for many years. An early shape from silhouette method was presented by Martin and Aggarwal [47] and has subsequently been refined by many other researchers. The approach relies on the following idea. The silhouette of the object in the image plane and camera location for each view forms a cone containing the object. See Fig. 2.12. All space outside of this cone must be outside of the object, and the cone represents a conservative estimate of the object shape. By intersecting the cones formed from many different viewpoints, the estimate of object shape can be refined. Different techniques for computing the intersection of cones have been proposed. Martin and Aggarwal considered a volume containing the object and uniformly divided such volume in sub-volumes. For each view, each sub-volume − or voxel − is examined to see if it is outside of the solid formed by the silhouette and the view point. If it is outside, the voxel is excluded from further estimates of the object shape. Subsequent research has improved on the efficiency of this approach by alternative data structures [68] for storing the in/out status of voxels. Standard surface extraction methods such as Marching Cubes [43] can be used to compute the final triangle mesh surface. It is also possible to model the cones directly as space enclosed by polygonal surfaces and intersect the surfaces to refine the object estimate, similar to the method used by Reed and Allen to merge range images [52].

Laurentini [40] introduced the idea of *visual hull*, i.e., the closest approximation to the object surface that can be estimated from n views. He also showed that the accuracy of a shape from silhouettes approach is limited. For instance, some concavities in the object are never observed from the silhouettes. Koenderink [35] studied the differential properties of occluding contours and the relationship between contours and local shape. Such analysis was extended by [23, 76]. In particular, the idea of using a series of silhouettes obtained from very small angular rotations of the object was explored. As illustrated by Cipolla and Blake [14] and Zheng [83], the depth of a point on the silhouette can be computed by analyzing the change in silhouette

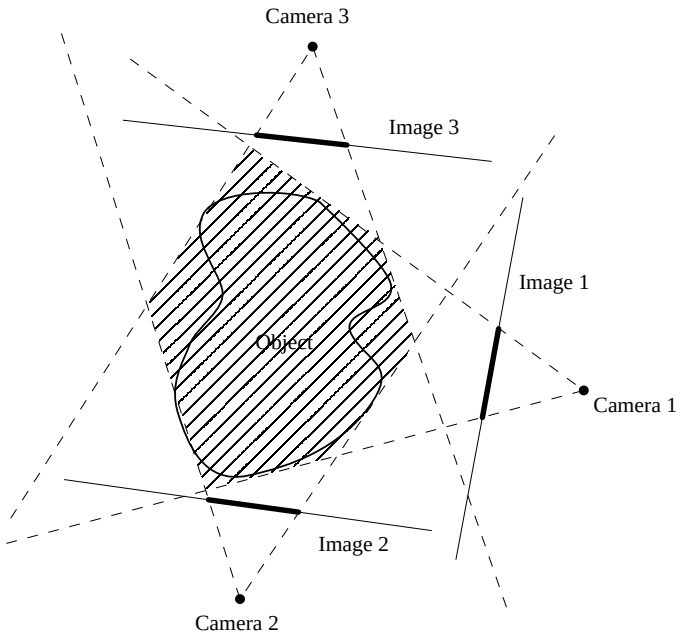


Figure 2.12: Shape from Silhouettes: The silhouette and camera location for each view forms a cone containing the object. The intersection of multiple cones is a conservative estimate of object shape.

for a small change in orientation. The set of 3D points obtained from many views form a cloud that can be integrated into a single surface mesh. The method presented by Zheng has the advantage of the unobserved areas that have unmeasured concavities being identified automatically. It has the disadvantage that many more silhouettes are required to estimate the object shape. The method is also not as robust as the cone intersection methods, because reconstruction errors inherent in meshing noisy points may result in holes or gaps in the surface. Alternative ways of exploiting silhouettes were proposed by [41, 19, 10, 65].

Because they are robust and conservative, volume-based space carving techniques similar to Martin and Aggrawal's original method have found success in low-end commercial scanners. The shape error caused by concavities not apparent in the silhouettes is often successfully masked by the use of color texture maps on the estimated geometry.

Although they are simple and relatively robust, shape from silhouettes approaches fail when they are unable to accurately segment the object from its background. Many systems use a backdrop of a solid, known color – i.e., the backdrops used in chroma-key systems for video compositing. This approach can fail when the object itself has the same color as the background. More frequently, it fails for objects with some specularity that reflect the backdrop into the direction of the camera view. Diffuse white objects may also reflect the color of the backdrop towards the camera through multiple self-inter-reflections. This reflection of the backdrop color can cause the segmentation to fail in two ways. The object boundary may be estimated as entirely inside the actual boundary, resulting in a general shrinkage. Areas in the middle of the object may be

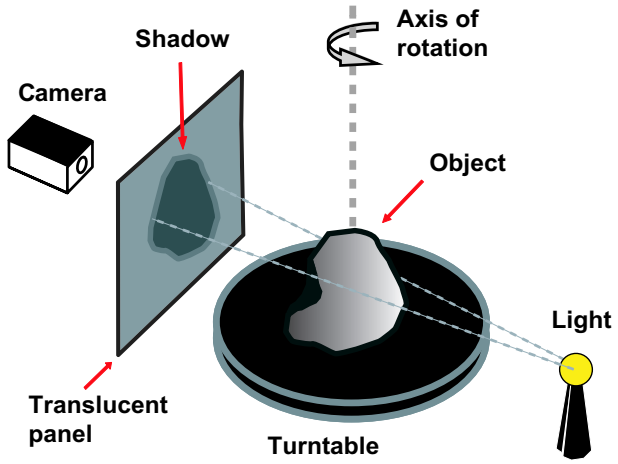


Figure 2.13: A novel setup for shape from silhouettes: A camera observes the shadow cast by a point light source on a translucent panel.

classified as backdrop. This results in the more serious error of tunnels in the object. A simple approach to correct tunnelling errors is to have the user inspect the images being segmented, and paint in areas that have been misclassified. Another approach is to use a large, diffusely emitting light source as the backdrop. This can often prevent areas in the middle of the object from being misclassified, but does not guarantee that areas near the silhouette edge that scatter light forward will be properly classified. It also prevents texture images from being acquired simultaneously with the silhouette images.

Recently, Leibe *et al.* [42] developed a shape from silhouettes approach that avoids the segmentation problem by using cast shadows. A ring of overhead emitters cast shadows of an object sitting on a translucent table. A camera located under the table records the shadow images. The emitter positions and shadows form the cones that are intersected to form a crude object representation. Because only one object pose can be used, the representation cannot be refined. The crude representation is adequate, however, for the remote collaboration application being addressed by Liebe *et al.*.

To avoid the segmentation problem inherent in many shape from silhouette systems, we adopt an approach similar to that of Leibe *et al.* [42]. We rearrange the set up, however, to allow for multiple object poses and better refinement of the object shape.

2.4.2 First Phase – Shape from Silhouettes

Our proposed new set up for shape from silhouettes is shown in Fig. 2.13. A point light source is placed in front of the object to be measured sitting on a turntable, casting a shadow on a translucent panel. A camera on the opposite side of the panel records the image of the shadow cast on the panel. The locations of the camera, light source, and panel relative to a coordinate system fixed to the turnable are found by calibration. In order

to be considered a “point” light source, the lamp simply needs to be an order of magnitude or more smaller than the object to be measured, so that the shadow that is cast is sharp. The lamp needs to have an even output, so that it does not cast patterns of light and dark that could be mistaken for shadows. The translucent panel is any thin, diffusely transmitting material. The panel is thin to eliminate significant scattering in the plane of the panel (which would make the shadow fuzzy) and has a forward scattering distribution that is nearly uniform for light incident on the panel, so that no images are formed on the camera side of the panel except for the shadow. The positions of the light source and camera are determined by making sure that the shadow of the object falls completely within the boundaries of the translucent panel for all object positions as the turntable revolves, and that the camera views the complete translucent panel.

By using a translucent panel, the camera views an image that is easily separated (e.g., by performing a k-means clustering) into two regions to determine the boundary intensity value between lit and unlit areas. Because the camera and panel positions are known, the shadow boundary can be expressed in the world coordinate system. The cone that fully contains the object is formed by the light source position and the shadow boundary. A volume can be defined that is initially larger than the object. Voxels can be classified as **in** or **out** for each turntable position. This classification can be done by projecting the voxel vertices along a line starting at the point light source onto the plane of the panel and determining whether they are in or out of the observed shadow. Voxels for which the result of this test is mixed are classified as **in**.

A more accurate estimate of the surface can be obtained by computing the actual crossing point for each in-out edge.

By using the projected shadow, problems such as the object having regions the same color as the background, or reflecting the background color into the direction of the camera are eliminated. The configuration also allows for some other interesting variations. Multiple light sources could be used for a single camera and panel position. One approach would be to use red, green and blue sources, casting different color shadows. In one image capture, three shadows could be captured at once. Another approach would be to add a camera in front, and use several light sources. For each turntable position, several shadow images could be captured in sequence by the camera behind the translucent panel, and several images for computing a detailed texture could be captured by the camera in front. See Fig. 2.14.

Either configuration for using multiple light sources would give information equivalent to using multiple camera views in a traditional shape from silhouettes set-up. The result of this is a more complete object description from each full rotation of the turntable, and a reduction in the number of positions of the object required to obtain a well-defined surface.

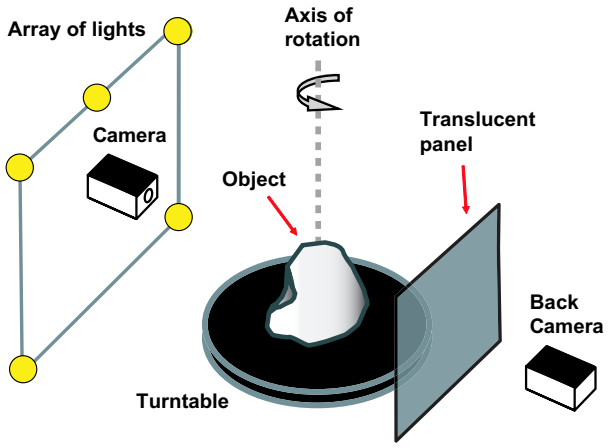


Figure 2.14: Alternative setup for shape from silhouettes and shape from self-shadowing: Shadow images on the translucent panel are obtained for multiple point light sources in front of the object and captured by a camera placed behind the translucent panel. At the same time a camera in front of the object is used to take images of the front of the object and the shadows it casts onto itself from the same light sources.

2.4.3 Combining Approaches.

It has become evident that to produce robust scanning systems it is useful to combine multiple shape-from-X approaches. A system is more robust if a shape estimated from shape-from-A is consistent with shape-from-B. An example is recent work by Kampel and Sablatnig [31] where cues from silhouettes and structured light are combined. An alternative approach is photo-consistency, as introduced by the work of Seitz and Dyer [62], Kutulakos and Seitz [38] and numerous extensions and/or alternative formulations [11, 2, 69, 6, 17, 24, 74]. The philosophical importance behind this work is summarized by the idea that a surface description is acceptable only if it is consistent with all the images captured of the surface. We use this basic concept in combining shape from silhouettes with shape from shadows. We refine our initial shape estimate obtained from shape form silhouettes with the conservative removal of material to generate shapes that are consistent with images of the object that exhibit self-shadowing.

2.4.4 Second Phase – Shadow Carving

As discussed in section 2.4.1, shape from silhouettes cannot capture the shape of some concave areas which never appear in object silhouettes. The new configuration described in section 2.4.2 does not overcome this limitation. Object concavities, however, are revealed from shadows the object casts on itself. In our second phase of processing, we analyze the object's self-shadowing and use shadow carving to adjust our surface estimated in the first phase.

Our implementation of shadow carving is organized in three parts: shadow detection, check for contradiction, and adjustment to resolve contradiction. First of all, we need to obtain multiple images with potential

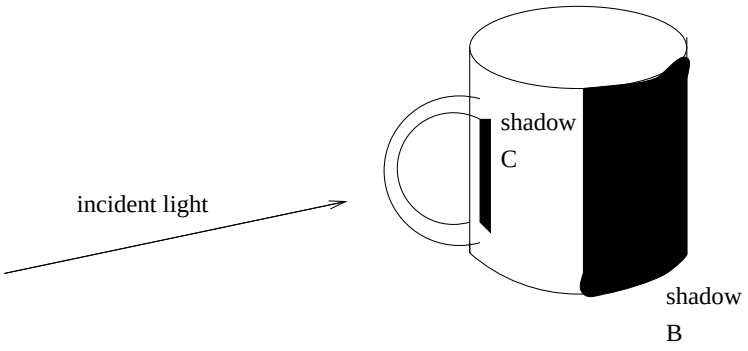


Figure 2.15: Not all shadows on an object indicate a contradiction with the current object estimate. For the mug, the shadow C cast by the handle and the attached shadow B are both explained by the current object estimate.

self-shadowing. We propose a hardware set up as shown in Fig. 2.1. A camera in front of the object is used to take images of the object and the shadows it casts onto itself from an array of light sources. We use these images to refine the object estimate. An alternative hardware configuration is shown in Fig. 2.14. Shadow images on the translucent panel are obtained for an array of light sources located in front of the object and captured by a camera placed behind the translucent panel. At the same, time a camera in front of the object is used to take images of the object and the shadows it casts onto itself from the same light sources. While all of the front and back images are taken in the same rotation of the turntable, the translucent panel images are processed first to obtain a first surface estimate (shape from silhouettes). We use the images obtained by the front camera in a second phase to refine this estimate (shadow carving).

Detecting shadows is not easy. In objects that have significant self-inter-reflection and/or spatially varying surface albedo, lit portions of the object often have low intensities – sometimes even lower than those attached to portions that do not have a direct view of the light source – but are lit by inter-reflection. A full solution to the shadow detection problem is not the objective of this work. In fact, it is permissible with our approach to avoid finding the exact shadow boundaries - we can misclassify shadow pixels as lit without jeopardizing the conservative property of our estimate. See section 2.3.4. Our goal then is to identify areas we are very certain to be in shadow region. At that end we make use of multiple light sources for each object position. We combine the images captured with these different light positions into a unique reference image by taking the maximum value for each pixel. Each image is then analyzed for areas that are dark, relative to the reference image, and also have an intrinsic low value of intensity. We select a threshold value that gives an estimation safely within the dark region for all of the images. The rest of our method is designed to make use of these conservative shadow estimates to continue to refine our initially conservative object estimate.

In the next step, we test whether the shadow regions thus identified can be explained by the current object estimate (check for contradiction). Notice that not all shadows observed on an object indicate that the initial

surface estimate is incorrect. Consider the case of a coffee mug shown in Fig. 2.15. Two shadows would be observed: the attached shadow B and the shadow cast by the handle C. Ray-tracing would show that both of these shadows are explained by the current surface. A ray to the light source in the attached area B would immediately enter the object itself, indicating the light source is not seen. A ray from area C would intersect the handle before reaching the source.

The problem remains of what to do to resolve unexplained shadows (adjustment to resolve contradiction). The surface estimate is conservative, so we cannot add material to the object to block the light source. We can only remove material. Removing material anywhere outside of the unexplained shadow will not form a block to the light source. The only option is to remove material from the unexplained shadow region. This can be done to the extent that the surface in that region is pushed back to the point that its view of the light source is blocked as explained in theorem 2.3.1.

As a final remark, in order for shadow carving to work, we assume that there are no features smaller than our image pixel size. See Section 2.3.4 for details. As noted in [51] and [26], this interpretation of shadows fails when there are features sharper than the image resolution.

2.5 Implementation

We have built a table-top system to test our approach. Our modeling software has two major components: space carving using the silhouette images and shadow carving using the images acquired with an array of light sources.

2.5.1 Hardware Setup

Our table-top setup is shown in Fig. 2.16. The system is composed of a *Kaidan MC-3* turntable, a removable translucent plane, a single 150 W halogen light source on the back of the object. Behind the movable panel, a rigid panel contains a *Fuji FinePix S1 Pro* digital camera, an array of five halogen light sources surrounding the camera and a *ShapeGrabber*^{TM} laser scanner which was used to obtain a ground truth. We capture 1536×2304 resolution images and crop out the central 1152×1728 region from each image. Control software allows us to automatically capture a series of N steps of images in rotational increments of $360/N$ degrees and turn the light sources on and off when necessary. The single halogen light source on the back of the object is used to project the shadow on the translucent plane. The translucent plane can be removed when necessary. It is made of frosted glass standing on a rigid support. A sheet of copier paper sandwiched between panes of glass can be used as well. For additional information about the setup and how to obtain the

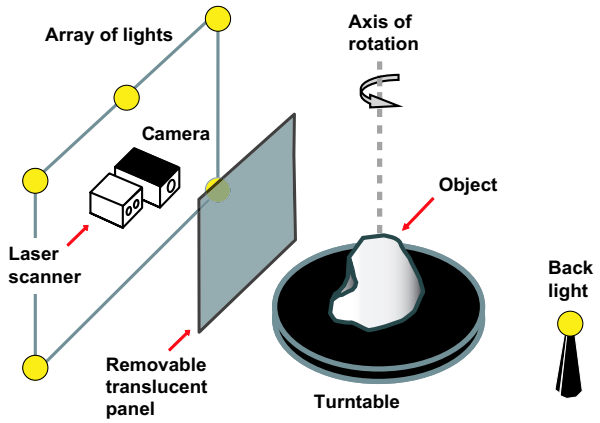


Figure 2.16: Setup of our proposed system. The system is composed of a turntable, a removable translucent plane, a single 150 W halogen light source on the back of the object. Behind the movable panel, a rigid panel contains a digital camera, an array of five halogen light sources surrounding the camera and a laser scanner. Notice that the laser scanner is only needed to acquire a ground-truth model of the object and is not required for our reconstruction algorithm.

ground truth see [18].

The camera is calibrated with respect to a coordinate system fixed to the turntable using a calibration artifact attached to the turntable and a Tsai camera calibration algorithm [75]. The location of laser scanner is calculated by scanning the artifact used to calibrate the camera. The location of the translucent panel and the position of the back light source are measured using the laser scanner. This is done for convenience - alternative calibration algorithms may be used as well. Since the panel surrounding the camera has known geometry, the location of the array of five light sources is known as well.

Because we use a single camera system, we need to take data in two full rotations of the turntable. In the first rotation, the translucent panel is in place and the silhouette images are obtained. These data are used by the space carving algorithm. In the second rotation of the turntable, the panel is removed without moving the object, and the series of five images from the five camera-side light sources are captured. These data are used by the shadow carving algorithm. [2]

Cropped images from the sample data acquired by our system are shown in Fig. 2.17. The left image shows the shadow cast by the object on the translucent panel when it is lit by the back light. The right image shows one of the five self-shadowing images captured for one object position. Even if the original images are captured in color, shadow carving method only requires gray scale. Color images can be used if a final texture mapping step is needed.

[2] A two camera system (one located behind the translucent panel and other one located, for instance, on the back - see Fig. 2.14) is more expensive but presents some advantages: i) only one rotation is needed, and the user is not required to step in and remove the panel; ii) it is possible to acquire many objects with the same calibration data.

Figure 2.17: Sample data from table-top system: shadow panel image on the left, and self-shadow image on the right.

2.5.2 Software – Space carving

Our goal is to recover the shape of the object from the silhouettes of the shadows cast into the translucent panel. Thus, the first step is to extract such shadow silhouettes. This task is very simple. We begin by doing k-means analysis on the images to determine the pixel intensity value dividing the lit and unlit regions for the images of the panel. The boundary of each shadow is then computed with sub-pixel accuracy. The estimation of the shadows is slightly dilated to account for errors in the calibration of the camera (e.g., errors in estimating the camera center and back light position).

After that the silhouettes are extracted, the space carving algorithm is organized as follows. A single volume that completely encloses the working space of the acquisition system is defined. In order to accelerate the computation, an octree scheme is used as proposed by Szeliski [68] and Wong [78]. The initial volume is subdivided in eight parts (voxels), each one marked as containing the surface of the object (**boundary** label). Each **boundary** voxel is projected into every shadow image. If there is at least one projection in which the voxel lies completely outside a shadow image than the corresponding voxel is marked as being outside the object (**out** label). If all the projections of a voxel lie completely inside the corresponding shadow images then the voxel is classified as being inside the object (labeled as **in**).

If some projections of a voxel partially overlap the shadow edges and some are fully inside other shadows, then the voxel keeps the **boundary** label. After all the voxels are inspected, those labeled **out** are discarded, those labeled **in** are kept but processed no further, and each **boundary** voxel is subdivided into eight new sub-voxels. All of the new voxels are processed as before to improve the estimate of the object volume.

When a suitable subdivision level is reached, each vertex of each **boundary** voxel is projected into the shadow images and classified as being **in** or **out** of the object. Additionally, all of the voxels that were labeled

Algorithm Shadow Carving

1. for each camera view
2. Threshold shadow images using reference image;
3. Render depth buffer using current surface;
4. for each shadow image;
5. Cast rays from each shadow pixel and mark contradictions;
6. Update depths for pixels in contradiction;
7. Update volume data using adjusted depth map;
8. Extract new surface estimate for full object;

Figure 2.18: Skeleton of the shadow carving algorithm.

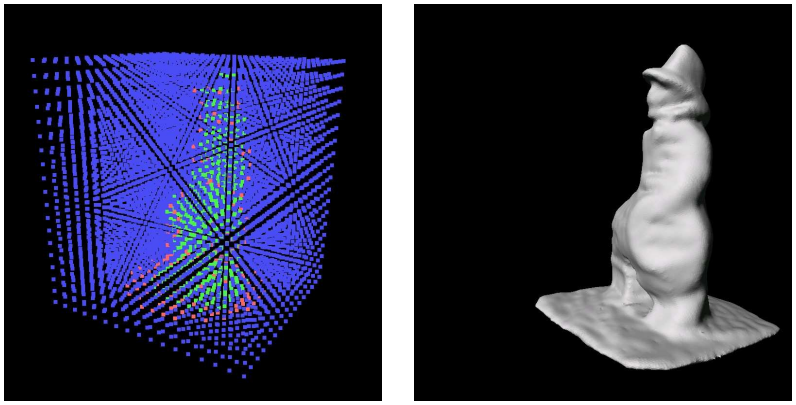


Figure 2.19: At the start of a shadow carving step, the current object estimation is represented by both a volumetric description (left), and a triangle mesh (right). The voxels of the volume are marked as inside (green), boundary (red), and outside (blue).

as **in** are subdivided to the finest level, and their vertices are labeled as **in**. Finally, all of the **boundary** and **in** voxels are saved to a file that provides the volume description used for the shadow carving algorithm.

2.5.3 Software – Shadow Carving

Our implementation of the shadow carving algorithm consists of three steps: shadow detection, test for contradiction, and surface update. These three operations are performed for each camera position, with the cumulative results for all positions $k-1$ carried forward to the processing of position k. The pseudocode of the algorithm is in Fig. 2.18. At the end of the processing for the $k-1$ position, the object is represented in two ways: as a voxel data structure, that describes its volume, and as a triangle mesh, that describes its surface and is extracted from the current volume using the Marching Cubes algorithm [43]. The voxel data structure maintains information for classifying voxels as completely inside, completely outside, or containing

Figure 2.20: In the first step of shadow carving, shadows are identified in a set of four images all taken from the same viewpoint, but with different light positions (first 4 images in the top row). The reference image is obtained by combining the images captured with different light positions into a new one by taking the maximum value for each pixel (right most image). The areas classified as shadow are superimposed in white over the original images (bottom row); the original intensity of the images is modified to highlight the detected shadows.

the surface of the object (see Fig. 2.19). We have implemented the algorithm in C++ using the OpenGL library as rendering engine.

Step 1 - shadow detection. Because a full solution to the shadow detection problem is not the object of this investigation, we use a simple approach in this implementation. We combine the images captured with five different light positions into a unique reference image by taking the maximum value for each pixel. Each image is then analyzed for areas that are dark relative to the reference image, and also have an intrinsic low value of intensity. We select a threshold value that give an estimation safely within the dark region for all of the images: the goal is to guarantee a conservative estimation of the shadows. Because we acquired objects with uniform albedo, we simply use the same threshold values across all the images of the same object. Examples are shown in Fig. 2.20.

Step 2 - test for contradiction. The next step is to test for whether points in observed shadow regions are in contradiction with our initial surface or not. We begin by rendering a depth image of our current surface estimate. This is done by simply reading back the depth-buffer after rendering the mesh of triangles that describes the object surface using OpenGL. The depth image has the same resolution as our captured shadow

images. An example of depth image is presented in Fig. 2.21. Higher values of intensity correspond to closer parts of the surface.

Figure 2.21: The surface estimate from the $k-1$ step is rendered as a depth image from the same viewpoint as the shadow images from the k step. The background pixels, in black, are set to the furthest possible depth value.

We then process each image for each view. For each pixel in the detected shadow region that has a non-background depth, we test if the ray from the corresponding point on the surface to the light source is blocked by the object. We call this test the *contradiction test.* If the ray is not blocked by the object than the shadow pixel is in contradiction and we need to update its depth value. Once that the depth value of every pixel in contradiction is updated, we obtain a new depth image which is consistent with the shadow image.

Algorithm Contradiction Test

```
1.   Initialize the point p to be the point whose
         projection in the image plane
         is the pixel in shadow we want test;
2.   while the projection of p is inside the shadow
3.       Move p along the line that connects p to the light;
4.       Project p into the image;
5.   Save the coordinate of the last projection;
6.   while p is not coincident with the light
7.       Move p along the line that connects p to the light;
8.       Project p on the image and get its depth d;
9.       if d is greater than the corresponding value
                 of the depth image
10.          return a non-contradiction flag;
11.  return a contradiction flag;
```

Figure 2.22: Contradiction test algorithm.

The algorithm that implements the contradiction test is described in Fig. 2.22. The algorithm takes as

input the current depth map, the shadow image as well as the coordinates of the pixel in shadow we want to test. Call p the point on the surface whose projection into the image plane gives the pixel in shadow we want to test. The point p is moved toward the light and projected into image plane (first while loop). This loop is repeated until the projection of p into the image plane falls outside the shadow region. Then the pixel corresponding to this projection is saved, and the algorithm keeps on moving p toward the light (second while loop). If p reaches the light without intersecting the object - i.e., the depth value of the projection of p is always smaller than the corresponding value of the depth map - then there is a contradiction [3]. The algorithm returns the saved value of the last pixel in shadow and a flag value indicating that the depth map can be updated. If p intersects the object then there is no contradiction and no need to update the depth map for the investigated pixel.

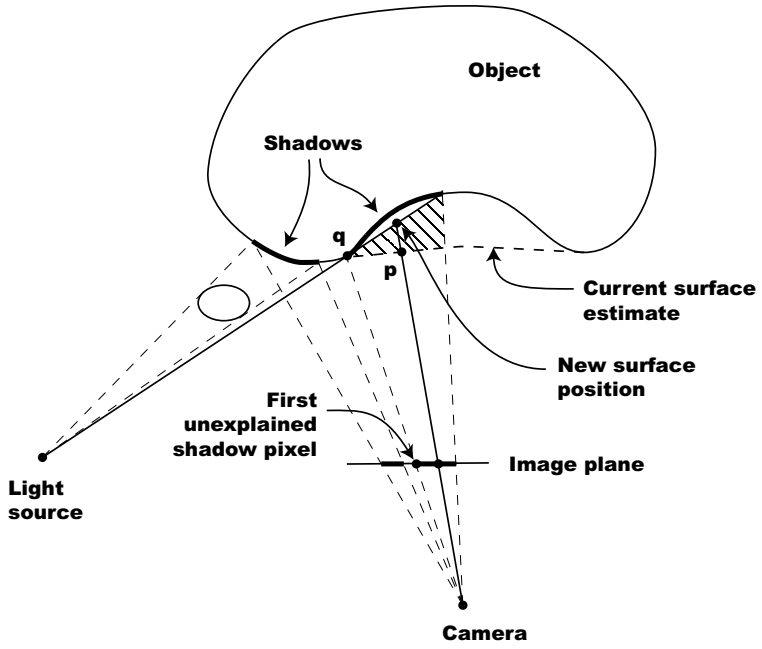


Figure 2.23: Adjustment of the position of a surface point when a contradiction is detected.

Step 3 - surface update. To resolve a pixel in contradiction for a given shadow region, the pixel's height in the depth map is adjusted according to the following algorithm. Consider the 2D slice shown in Fig. 2.23. Call *first unexplained pixel* the last pixel in shadow returned by the contradiction test. Call q its projection into the surface. The depth of each pixel in contradiction is then adjusted along the line of sight from the current camera view point and increased to the point it reaches the ray from the light source through q. This step can be repeated for each pixel in contradiction for a given shadow region. A new depth map is thus produced.

[3] Alternatively, for the contradiction test we could render a shadow map (i.e., a scene rendered from the light as if it was a camera) and save the depth map. Once the point in shadow is projected into the shadow map, the check would be if the point is further away than the closest value in the shadow map. The value of the last pixel in shadow has to be computed as in the previous case

The difference between the new and the old depth map corresponds (in the 2D slice) to the carvable areas introduced in Section 2.3. Proposition 2.3.1 guarantees that the depth map adjustment is always conservative.

After resolving all the pixels in contradiction for each shadow region for a given shadow image we obtain a new depth image. The new depth image is used to process the next shadow image acquired from the same viewpoint. The final depth map for the current camera view represents the adjustments for all shadow images from that viewpoint.

The last step for a camera view is to update the volume representation of the surface, so that a new full object representation can be extracted to carry forward to the next view. We test the voxel vertices to see if they lie in front of or behind the depth map surface and update the labeling of vertices which have been carved out. We use these updated values in the Marching Cubes processing to produce a new full surface representation. The surface representation can be rendered into a new depth map for the next camera view.

2.5.4 Software – Post Processing of the Surface

The final surface may be affected by small artifacts created during the reconstruction. These artifacts may be removed through a final postprocessing step. The post-processing is carried out by low-pass filtering the surface using the algorithm proposed by Taubin [70][71]. The filter parameters are chosen in order to remove the artifacts while preserving the geometry and the features of the model. In Section 2.6.1, we will address this issue in more details.

2.6 Experimental Results

We evaluate the performance of the space and shadow carving algorithms with both synthetic and real objects. As we shall discuss in Section 2.6.1, a commercial rendering package is used to simulate our system. This allows to test the correctness of the algorithms under a controlled environment, evaluate the performance with different configurations of lights and camera positions, and assess the accuracy of the reconstruction due to errors in the shadow estimates. In Section 2.6.2, we show examples of reconstruction of real world objects.

2.6.1 Experiments with Synthetic Objects

We have simulated our acquisition system shown in Section 2.5 with a commercial rendering package, *3ds Max 5.1*. The advantage of such experimental methodology is that an exact ground-truth is available for all of the geometrical quantities involved in the reconstruction process: the shape of the object we wish to scan, the object's silhouettes and the self-shadows. Thus, errors due to the calibration and misclassification

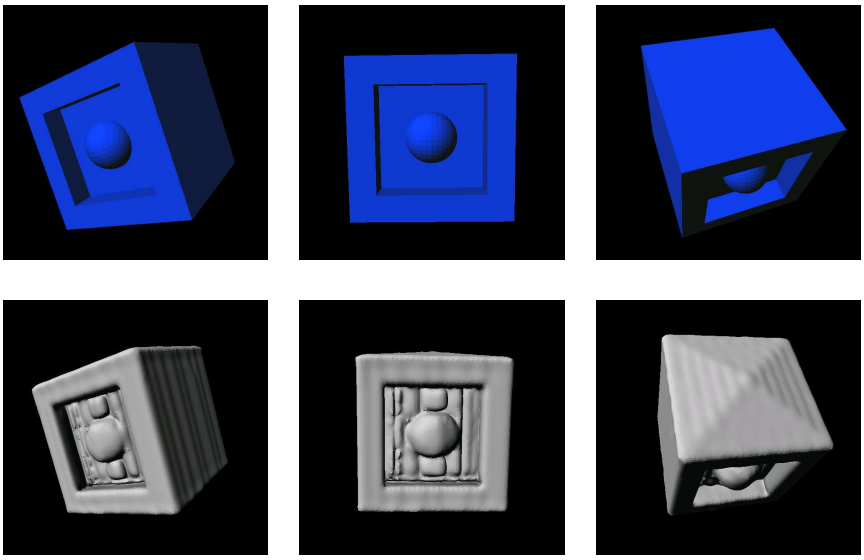


Figure 2.24: Top: three views of a synthetic cube. Bottom: results of shadow carving. 24 views and 4 lights are used in this experiment.

of shadow boundaries are ruled out. Since the virtual setup is more flexible and the experiments faster to run, such tests are also useful to study the effects of the light sources and camera configuration over the quality of the final reconstruction.

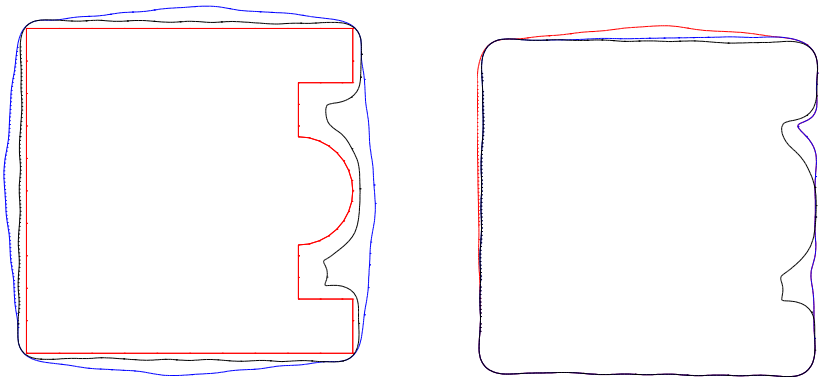
The simulated setup is similar to the scheme depicted in Fig. 2.16. Two different configurations are tested: the number of camera views (or, equivalently, the number of rotation steps of the turntable) are either 24 or 72; the number of light sources used to cast self-shadows are either 4 or 8. The configuration with 72 views corresponds to an angular rotation step of 5 degrees. Every image is rendered at a resolution of 640×480 pixels.

The configuration with 24 camera views and 4 lights is similar to the one in Section 2.5. Notice that since each silhouette needed for space carving can be easily extracted from a rendered image of the object, the translucent board to help silhouettes segmentation becomes superfluous. Finally, the shadow detection step (Section 2.5.3) is unnecessary since the ray-tracing algorithm allows to identify correctly all of the object's self-shadows in the images.

The first synthetic object used for the experiments is a simple cube with a cavity on one side, as shown in the first row of Fig. 2.24 [4].

The object was reconstructed using 24 views and 4 lights. The reconstruction is shown in the second Fig. 2.24. Notice that a portion of volume is removed from the cavity and that the new surface is still a

[4]The results shown in this figure and in the following ones were produced by Marco Andreetto's implementation of the shadow carving system.



(a) (b)

Figure 2.25: (a) Cross-sections of the real object (red); cross-sections of the result of space carving (blue); cross-sections of the result of shadow carving (black). (b) The progression of the approximations as we increased the number of viewpoints used for the reconstruction: red (8 viewpoints), blue (16) and black (24).

conservative estimate. However, several visual artifacts are clearly visible. This problem stems from the limited number of views and lights used for carving.

In Fig. 2.25 a cross section of the original object is presented. The cross section is defined by a horizontal plane passing through the center of the object and is depicted in red. The contours showing the estimate obtained by space carving are superposed in blue. The contours showing the estimate obtained by shadow carving are superposed in black. Notice that shadow carving yields an improvement both in the concavity (as expected) and along the side planes of the cube.

In the right side of Fig. 2.25, sections corresponding to partial improvements produced by shadow carving are presented: the red contour is obtained with 8 views (out of 24), the blue contour is obtained with 16 views (out of 24), the black contour corresponds to the final estimation.

The second synthetic object is a combination of cubes with a non-symmetric cavity on one side and few "extrusions" on the other ones. See the first row of Fig. 2.26. With only 24 views and 4 lights, shadow carving recovers the concavities only partially as shown in the second row of and in Fig. 2.27. Artifacts are left along Artifacts are left along the two sides of the main body of the object. Notice, however, that the estimation is still conservative and no volume is removed from the actual object.

A more accurate estimate can be obtained by increasing the number views from 24 to 72 and the number of lights from 4 to 8. The results are reported in the last row of Fig. 2.26. Notice that the volume has been almost completely removed from the cavity and the artifacts are greatly reduced in size and number. The cross sections of the reconstructed object with and without post-processing filtering (see Section 2.5) are shown in Fig. 2.28, respectively.

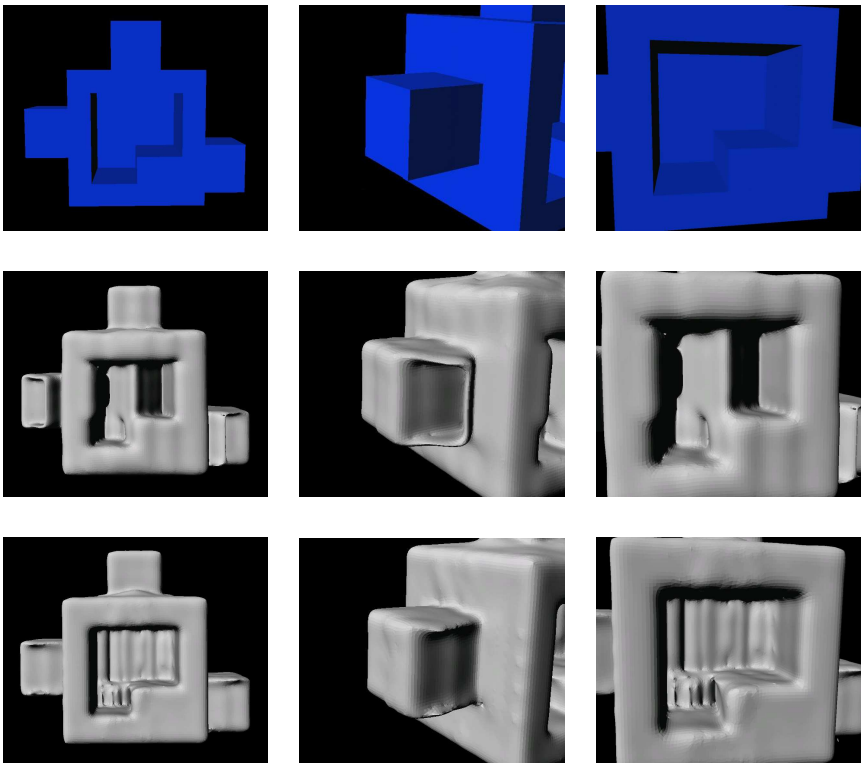


Figure 2.26: Top row: three views of the second synthetic object. Central row: results of shadow carving when 24 viewpoints and 4 lights are used. Bottom row: results of shadow carving when 72 viewpoints and 8 lights are used. Notice that the artifacts (e.g., see central concavity) dwindle away as the number of viewpoints and lights increase.

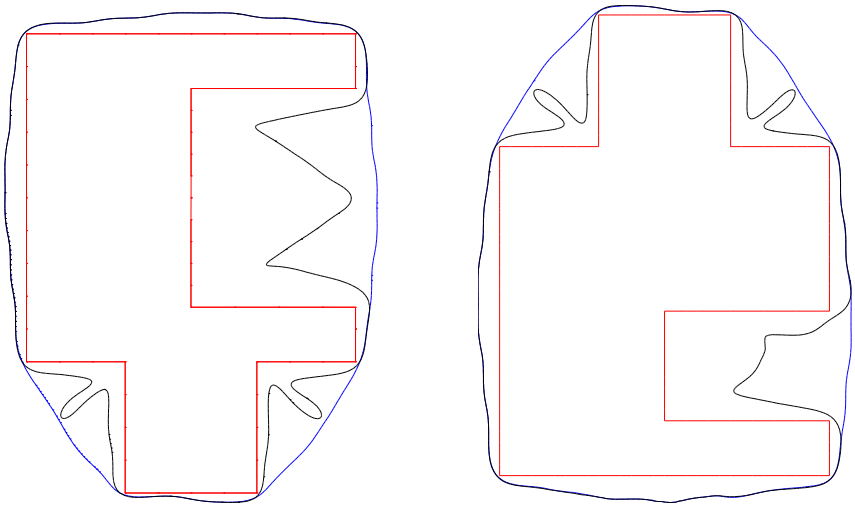


Figure 2.27: Two cross-sections of the actual object (red); cross-sections of the result of space carving (blue); cross-sections of the result of shadow carving (black). 24 viewpoints and 4 lights are used is this experiment.

Notice that since the artifact's scale (size) is smaller than the smallest feature of the object, we can filter out the artifacts as well as preserve the conservative property of the reconstruction.

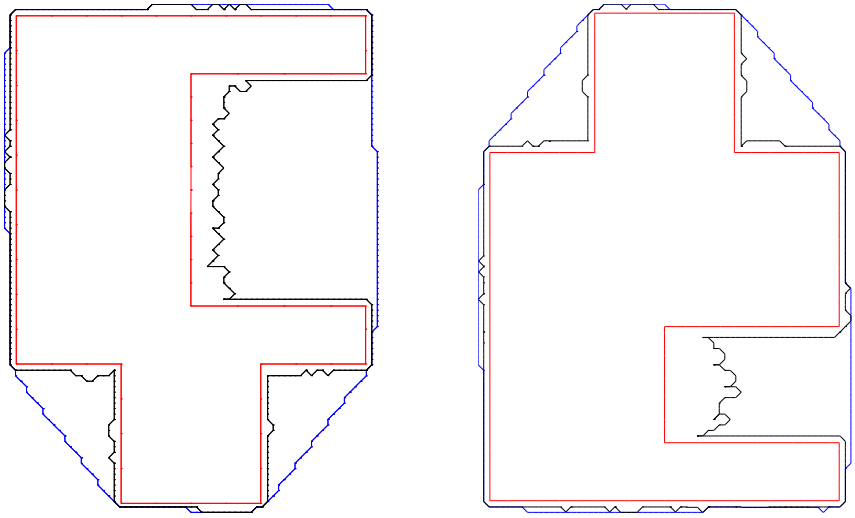


Figure 2.28: Two cross-sections of the actual object (red); cross-sections of the result of space carving (blue); cross-sections of the result of shadow carving (black). 72 viewpoints and 8 lights are used is this experiment. No post-processing filtering is carried out. Since the artifacts' size is smaller the smallest feature of the object, post-processing filtering may be useful to remove them. See Fig. 2.29

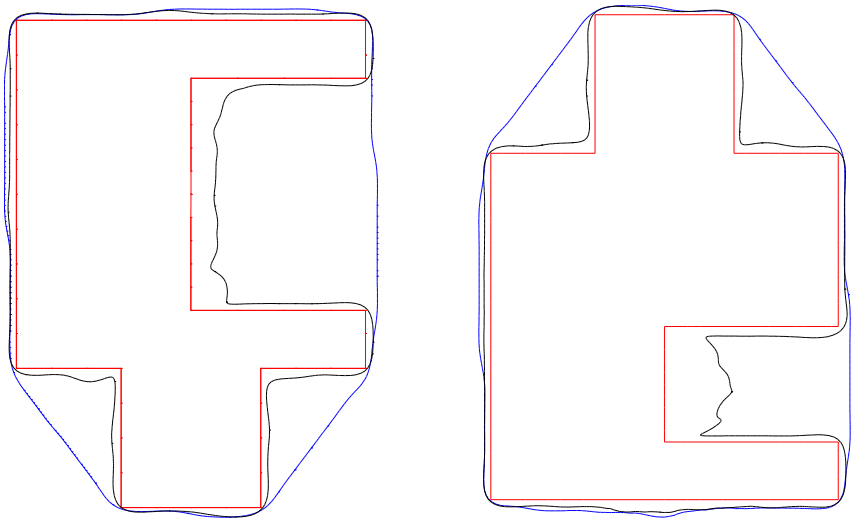


Figure 2.29: Cross-sections of the object in Fig. 2.28 after the post-processing filtering.

The simulated setup is also useful to study the effects of errors in detecting self-shadows. As predicted by our theory, if we violate the hypothesis that any lit pixel is not misclassify as shadow (see Proposition 2.3.1), the final reconstructed surface is not guaranteed to be conservative, i.e., portion of the actual object's volume can be carved out by the shadow carving algorithm.

To test this possibility we increased the size of the estimated shadows. The upper row of Fig. 2.30 shows examples of self-shadows cast over the surface of a sphere; the lower row shows the corresponding shadow estimates in black. The lit areas misclassified as shadows are in gray. As a result of such a misclassifica-

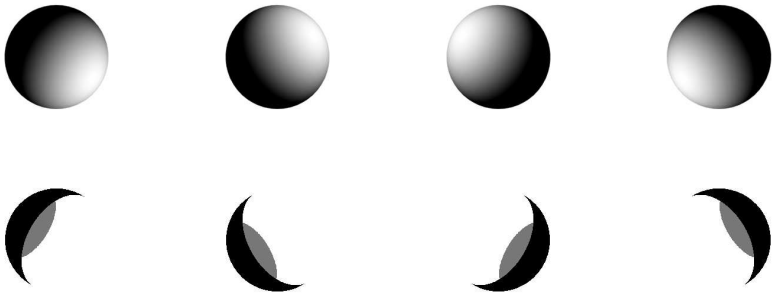


Figure 2.30: Examples of non-conservative shadow estimates. Upper row: examples of self-shadows cast over the surface of a sphere. Lower row: the corresponding shadow estimates are shown in black; the lit areas misclassified as shadows are in gray.

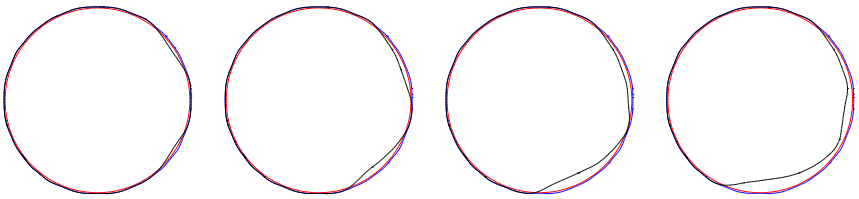


Figure 2.31: The effect of non-conservative estimate of the shadow. The figure shows the sections of the reconstructed sphere. As a result of the shadow misclassification, the algorithm starts carving out portions of the actual object, violating the assumption of conservative reconstruction. As shadow carving keeps on processing the next shadow images, the errors increase more and more.

tion, the algorithm starts carving out portions of the *actual* volume as shown in Fig. 2.31, most left panel. As shadow carving keeps on processing the next shadow images, the errors increase more and more. See remaining panels in Fig. 2.31.

2.6.2 Experiments with Real Objects

We have tested the carving hardware setup presented in Section 2.5 and used our shadow carving algorithm to scan several objects. Image resolution was 1152×1728 pixels. Our experiments were designed to test whether conservative estimates, as promised by theorem 2.3.1, may be obtained in a practical setting with noisy images and uncertain shadow detections.

Left panel of Fig. 2.32 shows the first object we reconstructed - a small sculpture of a dog. The central panel shows the results of shape from silhouettes. Notice that the algorithm fails at reconstructing the concavities of the object. The right panel shows results of shadow carving. Portions of the concavities are carved out although some artifacts are still visible. These are due to limited number of lights and viewpoints used to carve the model.

To assess the accuracy of our results, we have used the laser scanner to capture the 3D shape of the object.

Figure 2.32: Left: captured image of the dog. Center: results of space carving. Right: results of shadow carving with 4 lights and 24 views.

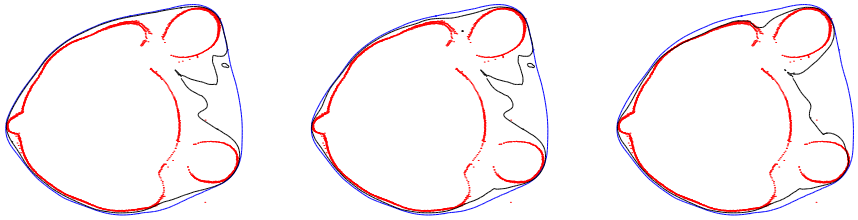


Figure 2.33: Cross-sections of the model of the dog reconstructed with the laser scanner (red), space carving (blue) and shadow carving (black). From left to right: progression of the approximations as we increased the number of viewpoints used for the reconstruction (8, 16 and 24 viewpoints respectively).

Such reconstructed models are visualized as a cloud of points rather than a triangulated mesh. As described in Section 2.6.1, we intersect a horizontal plane with the model to produce cross-sections, i.e., 2D contours.

In Fig. 2.33 cross-sections obtained from the laser scanner (red), space carving (blue), and shadow carving (black) are presented. The progression of approximations obtained by shadow carving are shown from left to right (8 , 16 and 24 views respectively).

The second object - a sculpture of a fireman - reconstructed by our carving system is presented in the left column of Fig. 2.34. The corresponding results of space carving and shadow carving are shown in the center and right columns respectively. Details of the reconstruction (Fig. 2.35) highlight the improvement of shadow carving over space carving. The cross-sections (Fig. 2.36 from left to right) show the progressive improvement of shadow carving.

In the left and right panels of Fig. 2.37 the reconstructed models (obtained by space and shadow carving respectively) are rendered from two camera views. A texture is superimposed to the models (texture-mapping). Notice that the model obtained by shadow carving is visually more compelling than the one by space carving.

A third example of reconstruction is presented in Fig. 2.39.

Figure 2.34: Upper row: two views of the fireman. Middle row: results of space carving. Bottom row: results of shadow carving.

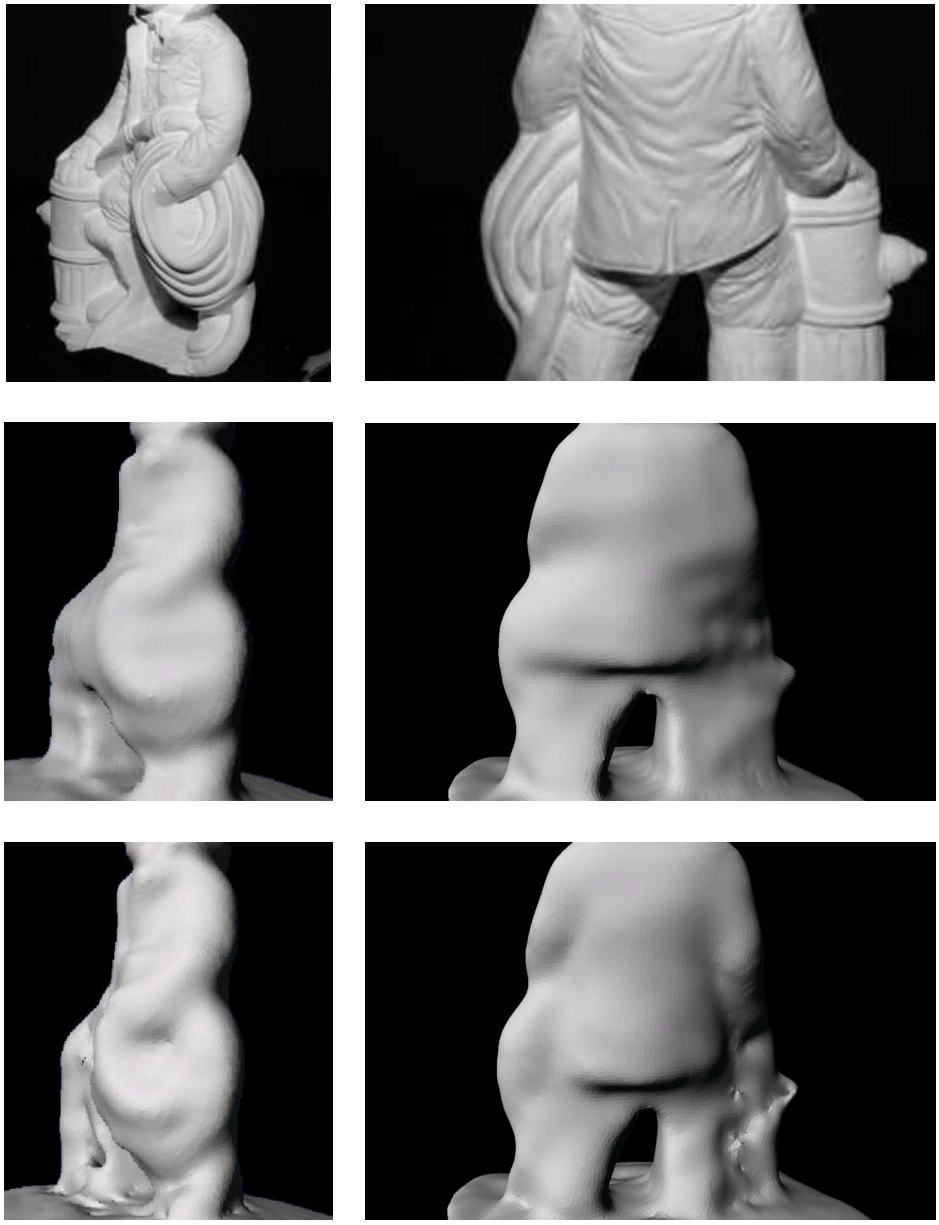


Figure 2.35: Details of the fireman. Upper row: two views of the original object. Middle row: results of space carving. Bottom row: results of shadow carving.

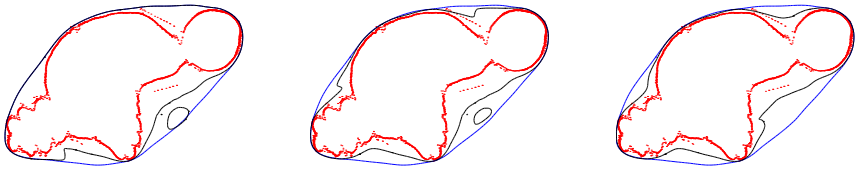


Figure 2.36: Cross-sections of model of the fireman reconstructed with the laser scanner (red), space carving (blue) and shadow carving (black). From left to right: progression of the approximations as we increased the number of viewpoints used for the reconstruction (8, 16 and 24 viewpoints respectively).

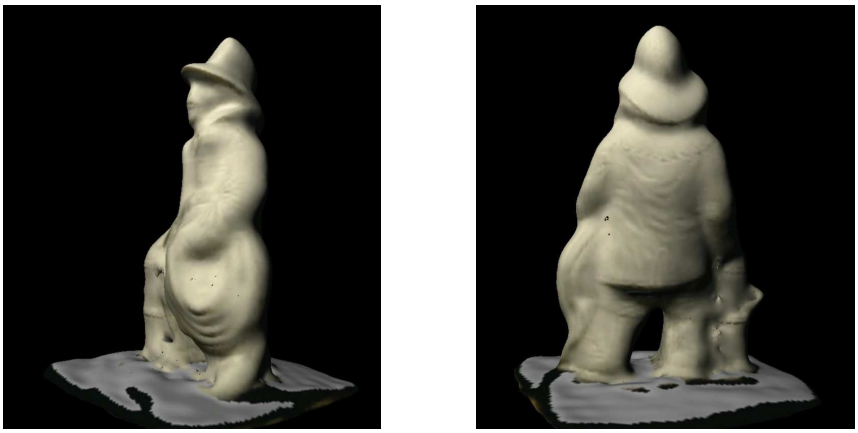


Figure 2.37: Reconstruction with texture mapping: the object's textures are added to the model captured with shadow carving.

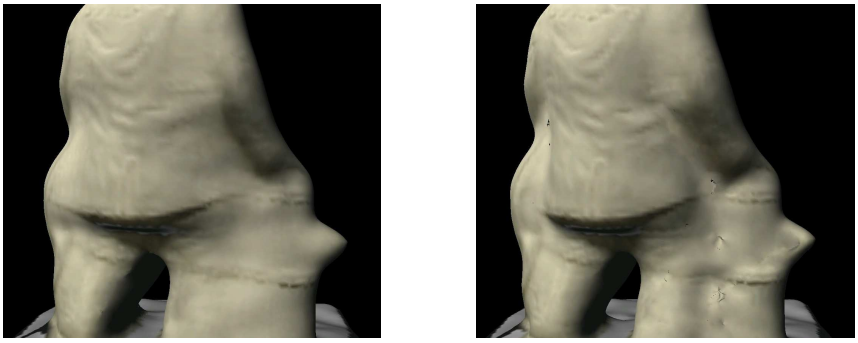


Figure 2.38: Comparison of the reconstruction with texture mapping. Left: the object's textures are added to the model captured with space carving. Right: the object's textures are added to the model captured with shadow carving.

2.6.3 Discussion

Our experiments show that the theoretical framework we proposed and analyzed in section 2.3 yields a practical algorithm which can improve upon the well-known space carving methods by carving into concavities of the object. Theorem 2.3.1 guarantees that the sequence of estimates produced is conservative and our experiments show that this is achievable in practice.

From our experiments it is evident that most concave areas have been carved only partially. This is due to two main reasons. Firstly, we used a small number of views and lights. This was predicted by our experiments with synthetic objects: the reconstruction obtained with 72 views and 8 lights is significantly more accurate than the one with 24 views and 4 lights only. Finding the optimal number and distribution of lights and views is clearly an important open problem. Secondly, our simple and overly conservative shadow detection algorithm reduces the amount of information available to the shadow carving algorithm. The reconstruction

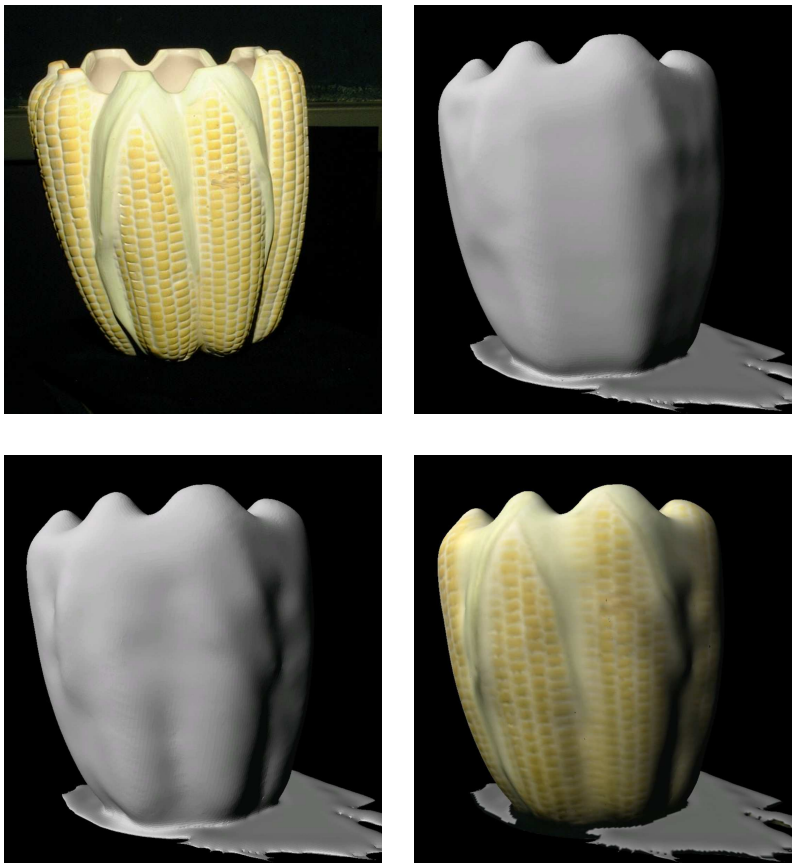


Figure 2.39: Upper left: image of the vase. Upper right: results of space carving. Lower left: results of shadow carving. Lower right : textured model

in Fig. 2.39 is an example. Detecting shadows in the vase's concavities is very hard: the edge are smooth, the concavities are brighter due to inter-reflections and some areas have a darker albedo. As a result only few shadows are detected and the shadow carving ability to remove volume is severely impaired. Detecting shadows accurately and yet preserving the conservative property is clearly another important open problem.

2.7 Conclusions

We presented an analysis of the problem of refining a conservative estimate of an objects shape by observing the shadows on the object when it is lit by a known point light source. We proved that a well defined portion of volume can be removed in a conservative manner from the current object estimate. We showed that this insight leads to an algorithm that can work correctly on real images. We called this method *shadow carving*.

Our main assumption is that one can estimate shadows conservatively - i.e., a shadow may not be detected but whatever is labeled as shadow is indeed a shadow. No assumptions about the object topology are neces-

sary. Tangent plane discontinuities over the objects surface have been supposed to be detectable. We showed that shadow carving improves previous work on shape from shadow in that it is more robust with respect to the classification of shadow regions and is not restricted to 2.5D terrain surfaces, but rather it may be applied to measuring the objects in the round.

In order to validate our theory, we have implemented a reconstruction system that combines information from silhouettes and shadows. The new system uses inexpensive digital cameras and lamps. Our experiments with real and synthetic objects confirms that the property of conservative carving is achievable in practice and show that shadow carving produces a much better surface estimate that shape from silhouettes alone. This improved estimate is suitable for further refinement by any shape estimation method that works well in local regions of the surface.

Chapter 3

Computational Analysis for 3D Reconstruction of Specular Surfaces

3.1 Introduction and Motivation

Estimating the 3D shape of physical objects is one of most useful functions of vision. Texture, shading, contour, stereoscopy, motion parallax and active projection of structured lighting are the most frequently studied cues for recovering 3D shape. These cues, however, are often inadequate for recovering the shape of shiny reflective objects, such as a silver plate, a glass goblet or a well-washed automobile, since it is not possible to observe their surfaces directly, rather only what they reflect.

Yet, the ability of recovering the shape of specular or highly reflective surfaces is valuable in many applications such as industrial metrology of polished metallic and plastic parts, medical imaging of moist or gelatinous tissues, digital archival of artworks and heritage objects, remote sensing of liquid surfaces, and diagnostic of space metallic structures, among others.

Although specular surfaces are difficult to measure with traditional techniques, specular reflections present an additional cue that potentially may be exploited for shape recovery. A curved mirror produces 'distorted' images of the surrounding world. For example, the image of a straight line reflected by a curved mirror is, in general, a curve. It is clear that such distortions are systematically related to the shape of the surface. Is it possible to invert this map, and recover the shape of the mirror from its reflected images?

The general 'inverse mirror' problem is clearly under-constrained; by opportunely manipulating the surrounding world, we may produce a great variety of images from any curved mirror surface. This inverse problem may become tractable under the assumption that the structure of the scene is known.

3.1.1 Proposed Approach and Summary of the Results

In this study, we assume that both the camera (modelled as a perspective camera) and the scene are calibrated. We ignore the contribution of visual cues such as shading and texture and we focus on the geometry relating a scene and its corresponding reflected image. We use this relationship to estimate the local shape of the mirror surface.

We start from the observation that the mapping from a scene line to its reflected curve in the camera image plane (due to mirror reflection) not only changes “orientation” and ”curvature” of the scene line, but also ”stretches” its length, modifying the local scale of the scene line (see Figure 3.1). We first analyze this map and derive analytical expressions for the local geometry in the image (namely, first- and second-order derivatives of the reflected image curve at any given point) as a function of the position and shape of the mirror surface . We then generalize our results to a planar scene of arbitrary texture and describe the mapping between a local planar scene patch to the corresponding reflected patch in the image plane as a function of the local parameters of the mirror surface. Finally, we explore the inverse problem in three settings.

First, we show that by exploiting position and orientation measurements of (at least) three reflected curves intersecting at an examined point, it is possible to recover the local geometry up to first-order (second-order up to one free parameter). An example is shown in Figure 3.5. Second, we show that by exploiting local position, orientation and local scale measurements of (at least) two reflected curves intersecting at a point, the surface geometry at such a grid point can be recovered up to third-order accuracy. An example is shown in Figure 3.10. Third, we demonstrate that scale and orientation measurements may also be extracted from the reflection of a planar scene patch of arbitrary geometry. We prove that local surface shape can be estimated if the location of (at least) three arbitrary points is available within a neighborhood of the reflected scene patch. These results are summarized in Table 3.1.

3.1.2 Previous Work and Our Contribution

Previous authors have used highlights as a cue to infer information about the geometry of a specular surface. Koenderink and van Doorn [36] qualitatively described how pattern of specularities change under viewer motion. This analysis was extended by Blake *et al.* to stereoscopic vision [4, 3]. Additionally, Zisserman *et al.* [84] investigated what geometrical information can be obtained by tracking the motion of specularities. Other approaches were based on the idea of modelling specular reflections with reflectance maps [28] or introducing them in the contest of photometric stereo [30]. Oren and Nayar [49] performed an analysis on classification of real and virtual features, and developed an algorithm recovering the 3D surface profiles traveled by virtual features. Zheng and Murata [82] developed a system where a rotating specular object is

Measurements	**Surface quantities up to 3^{rd}-order**				**DOF**	**Further details**
	Point	Normal	2^{nd}-order param.	3^{rd}-order param.		
Point position	s	s	a, b, c	e, f, g, h	8	Section 3.2.1
Point position + 2 orientations	s	s	r	e, f, g, h	6	Section 3.5.2
Point position + 3 orientations	✓	✓	r	e, f, g, h	5	Section 3.5.3.1 Section 3.5.3.4
Point position + 3 orientations + 4 curvatures	✓	✓	r	r	1	Section 3.5.3.7
Point position + 2 orientations + 2 scales	✓	✓	✓	✓	0	Section 3.5.4.1 Section 3.5.4.2

Table 3.1: Summary of the results. The first column shows different types of measurements available in the image plane. Each type of measurement enables the local reconstruction of the surface up to different degrees of freedom (which are shown in column 6). Columns $2-5$ indicate the corresponding free parameters which cannot be estimated given that amount of measured information. The free parameters are s, a, b, c, e, f, g, h and will be introduced in the next sections. A check mark (✓) indicates that all quantities may be recovered. For example (see third row), given the measurement of the position of one point and the orientations of three curves at that point, surface position and orientation can be fully recovered, curvature can be recovered up to one unknown (r), and third surfaces parameters (e, f, g, h) cannot be estimated at all. The last column points to further details.

reconstructed by illuminating it by extended lights and then analyzing the motion of the highlight stripes. Halsead *et al.* [25] proposed a reconstruction algorithm where a surface global model is fitted to a set of normals obtained by imaging a pattern of light reflected by specular surface. Their results were applied to interactive visualization of the cornea. Swaminathan *et al.* [66] presented an in-depth analysis of caustics of catadioptric cameras with conic reflectors. The same authors [67] gave insights on how to design the shape of mirrors in catadioptric imaging systems. Perard [50] and Tarini *et al.* [45] proposed a structured lighting technique for the iterative reconstruction of surface normal vectors and topography. Solem *et al.* [64] formulated the problem in a variational framework, using an implicit level set representation of the surface. Wang *et al.* [77] proposed a scanning system to recover fine-scale surface shape using a curved mirror to view multiple angles in a single image. Finally, Bonfort *et al.* [7] presented a novel voxel-based approach in the spirit of multiple view (space carving) algorithms. The main limitations of these studies are the necessity of having i) some knowledge on shape and position of the object; ii) multiple images under different condition of the illuminant; and iii) dedicated hardware equipment.

Our method departs from previous work in several aspects. First, we work on the static monocular case and explore the amount of information available from a single image. As we will show in Section 3.6.2, with only one image we can calibrate the camera, the scene and recover information about an unknown specular shape (the car fender, in the example of Figure 3.19). Second, our reconstruction scheme requires

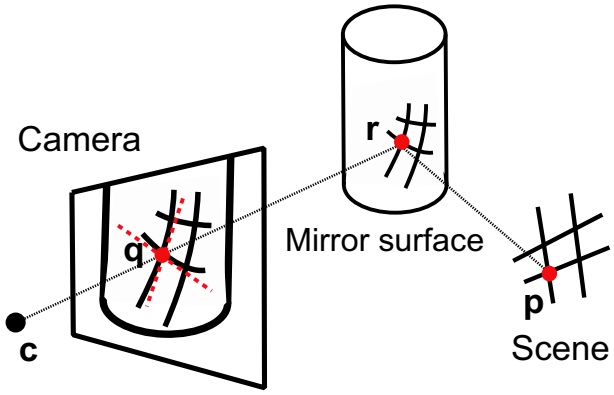


Figure 3.1: Geometry of the specular reflection

a simple setup and minimal hardware: a digital camera and a planar textured board. Finally, our analysis is local and differential rather than global and algebraic. For the first time, we provide an explicit differential relationship between the local structure of a scene patch, its corresponding reflection in the image, and the local shape of the specular surface (up to third-order). This chapter gathers and generalizes our own work presented in [58, 59, 54, 55], in that it provides: i) a complete first- and second-order differential description of the mapping between the local scene patch and the corresponding reflection in the image; in particular, the reconstruction schemes presented in [58, 59, 54, 55] are reframed in the novel formulation; and ii) a study of degenerate conditions of this mapping.

3.1.3 Chapter organization

The remainder of this chapter is organized as follows: we formulate the problem and introduce the motivation in Section 3.2; in Section 3.3, we present the direct problem and analyze how the surface geometry affects the image measurements using a differential approach; in Section 3.4, we present several properties of the reflection mapping and discuss its degenerate conditions; in Section 3.5, we apply the analytical expressions derived in Section 3.3 to study the inverse problem and present a new technique to estimate surface position and shape from various image measurements; in Section 3.6, we describe a practical algorithm for surface local reconstruction and validate our theoretical results with both numerical simulations and experiments with real surfaces; we finally discuss our findings in Section 3.7 and propose a number of issues for future research.

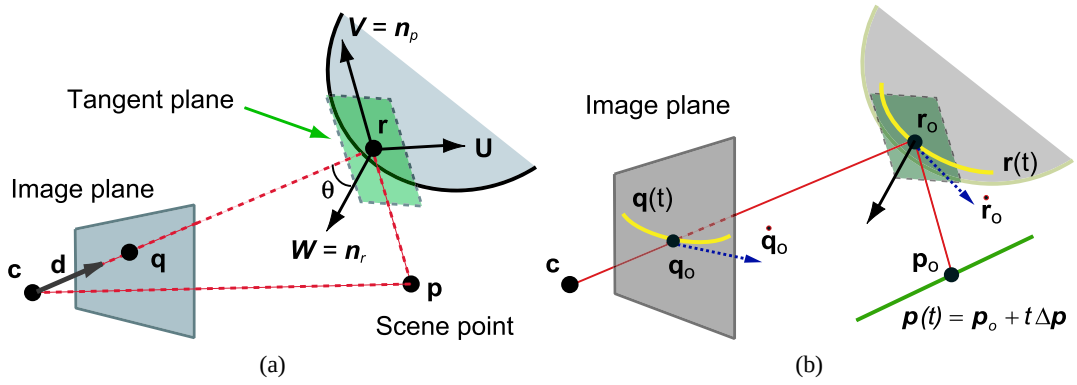


Figure 3.2: **Specular reflection geometry**

3.2 Problem Formulation

Our geometric setup is depicted in Figure 3.1. For now, we assume that a calibrated scene is composed of a planar pattern of intersecting lines. We will consider the general case of arbitrary planar scene in Section 3.5.4.5. The scene is reflected off an unknown smooth mirror surface and the reflection is observed by a calibrated camera. Our goal is to obtain local geometrical information of the mirror surface around $\mathbf{r}$ by analyzing the deformation produced upon the pattern of lines intersecting at point $\mathbf{p}$.

In Section 3.2.1, we introduce the notation and study the simple case of one known point reflected off the mirror surface. We show in Proposition 1 that, given the measurement of the reflected point in the image, the orientation of the surface at the reflection point is known up to one distance parameter. In Section 3.2.2, we introduce a new reference system to give a simpler surface representation. In Section 3.2.3, we define the mapping between a scene line and the corresponding reflected image curve. We conclude the section by sketching out our proposed reconstruction scheme.

3.2.1 Notation and Basic Geometry of Specular Reflections

Points and vectors (i.e., difference of points) in $3D$ are denoted by a bold lower case letter[1] (e.g., $\mathbf{x} = [x\ y\ z]^T$). Tensors, matrices and $4D$ vectors are all denoted by a bold capital letter (e.g., $\mathbf{X}$). Let $\mathbf{c}$ be the center of projection of the camera. The image plane is positioned l distance units in front of $\mathbf{c}$, perpendicular to the view direction $\mathbf{v}$. Given a scene point $\mathbf{p}$, let $\mathbf{q}$ be the image of $\mathbf{p}$ observed on the image plane through a specular reflection on the mirror surface at $\mathbf{r}$ (see Figure 3.2a). Since we assume that the camera and the scene pattern are calibrated, $\mathbf{p}$ and $\mathbf{q}$ are known, whereas $\mathbf{r}$ and the unit normal $\mathbf{n}_r$ to the surface at $\mathbf{r}$ are unknown. It follows from the perspective projection constraint that the point $\mathbf{r}$ must belong to the line defined by $\mathbf{c}$ and

[1] Notice that we use the same notation for points and vectors. This is done to improve readability.

$\mathbf{q}$, resulting in the following relationship:

$$\mathbf{r} = \mathbf{c} + s\,\mathbf{d}, \tag{3.1}$$

where the unit vector $\mathbf{d} = (\mathbf{q} - \mathbf{c})/\,||\,\mathbf{q} - \mathbf{c}\,||$ is parallel to the line of sight, and $s = ||\,\mathbf{r} - \mathbf{c}\,||$ is the distance from $\mathbf{c}$ to $\mathbf{r}$. With $\mathbf{c}$ fixed and $\mathbf{q}$ measured, the surface position at $\mathbf{r}$ is completely determined by a single distance parameter s. Furthermore, let us call the plane defined by $\mathbf{p}$, $\mathbf{r}$ and $\mathbf{c}$ the *principal plane*, let $\mathbf{n}_p$ be its unit normal vector and let θ denote the reflection angle at $\mathbf{r}$. Then:

Proposition 1 The unit normal vector $\mathbf{n}_r$ and the reflection angle θ are completely determined by the position parameter s.

Proof: The geometry of our setup satisfies 2 basic constraints:

1. By the geometry of specular reflection, the incident vector $\mathbf{r} - \mathbf{p}$ and the reflected vector $\mathbf{r} - \mathbf{c}$ are co-planar with the normal vector $\mathbf{n}_r$ and the plane they define is the principal plane, namely

$$\langle \mathbf{n}_r, \mathbf{n}_p \rangle = 0, \tag{3.2}$$

   where $\langle , \rangle$ denotes the inner product of two vectors.

2. Furthermore, the angle between the incident vector and $\mathbf{n}_r$ must be equal to the angle between the reflected vector and $\mathbf{n}_r$, namely

$$\langle \mathbf{n}_r, \mathbf{d} \rangle = \left\langle \mathbf{n}_r, \frac{\mathbf{r} - \mathbf{p}}{||\,\mathbf{r} - \mathbf{p}\,||} \right\rangle, \tag{3.3}$$

   which leads to

$$\left\langle \mathbf{n}_r, \mathbf{d} - \frac{\mathbf{r} - \mathbf{p}}{||\,\mathbf{r} - \mathbf{p}\,||} \right\rangle = 0. \tag{3.4}$$

By combining Eq. (3.2) and Eq. (3.4), we can express $\mathbf{n}_r$ as a function of s (up to a sign) as follows:

$$\begin{aligned} \mathbf{n}_r &= \left(\mathbf{d} - \frac{\mathbf{r} - \mathbf{p}}{||\,\mathbf{r} - \mathbf{p}\,||} \right) \times \mathbf{n}_p \\ &= \frac{\mathbf{p} - \mathbf{c} - (s - ||\,\mathbf{p} - \mathbf{c} - s\,\mathbf{d}\,||)\,\mathbf{d}}{||\,\mathbf{p} - \mathbf{c} - s\,\mathbf{d}\,||} \times \mathbf{n}_p. \end{aligned} \tag{3.5}$$

Similarly, the reflection angle θ can be parameterized by s as well by noting that

$$\cos 2\theta \;=\; 2\cos^2\theta - 1 \;=\; \left\langle \mathbf{d}, \frac{\mathbf{r}-\mathbf{p}}{||\,\mathbf{r}-\mathbf{p}\,||} \right\rangle,$$

which yields

$$\cos\theta \;=\; \frac{\sqrt{2}}{2}\sqrt{\frac{s-\langle \mathbf{d}, \mathbf{p}-\mathbf{c}\rangle}{||\,\mathbf{c}+s\mathbf{d}-\mathbf{p}\,||}+1}. \tag{3.6}$$

■

3.2.2 Reference System and Surface representation

Following Blake [3], we now introduce a new reference system to give a simpler surface representation and a more compact relationship between higher-order local surface parameters and the remaining geometrical quantities. We call this new reference system the *principal reference system* $[\mathbf{u}\,\mathbf{v}\,\mathbf{w}]$. The principal reference system is centered at $\mathbf{r}$ and $\mathbf{w} = \mathbf{n}_r$, $\mathbf{v} = \mathbf{n}_p$, and $\mathbf{u} = \mathbf{v} \times \mathbf{w}$. Given an arbitrary point $\mathbf{a}$ represented in a reference system $[\mathbf{x}\,\mathbf{y}\,\mathbf{z}]$ centered in $\mathbf{c}$, its corresponding coordinates $\mathbf{a}'$ in $[\mathbf{u}\,\mathbf{v}\,\mathbf{w}]$ can be obtained by a transformation $\mathbf{a}' = \mathbf{R}^{\mathrm{T}}(\mathbf{a}-\mathbf{r})$, where $\mathbf{R} = [\mathbf{n}_p \times \mathbf{n}_r\; \mathbf{n}_p\; \mathbf{n}_r]$. Proposition 1 implies that this transformation is a function of the unknown parameter s. Therefore, the choice of principal reference system does not introduce any other unknown variables. From now on, we will work in this principal reference system unless otherwise stated.

In the principal reference system, the normal of the surface at the origin is $\mathbf{w}$ and the tangent plane to the surface there is the plane defined by $\mathbf{u}$ and $\mathbf{v}$. Thus, the surface around $\mathbf{r}$ can be written in the *special Monge form* [15], yielding

$$\begin{aligned} w \;=\; & \frac{1}{2!}(a\,u^2 + 2c\,uv + b\,v^2) \\ & + \frac{1}{3!}(e\,u^3 + 3f\,u^2v + 3g\,uv^2 + h\,v^3) + \cdots, \end{aligned} \tag{3.7}$$

where u, v, w are the coordinates of the special Monge form in the principal reference system. a, b, c and e, f, g, h are the second-order and third-order surface parameters around $\mathbf{r}$, respectively. Accordingly, we refer to the distance parameter s as the first-order surface parameter since it determines the position and normal of the surface. Our goal is to recover these surface parameters around $\mathbf{r}$ using quantities that are known or measurable. Note that $\mathbf{c}$, $\mathbf{q}$, $\mathbf{p}$ are known by assuming calibrated camera and scene.

3.2.3 Differential Approach

Let $\mathbf{p}_0 (= [\mathbf{p}_{0u}\ \mathbf{p}_{0v}\ \mathbf{p}_{0w}]^{\mathrm{T}})$ denote a scene point in 3D space. A line $\mathbf{p}$ passing through $\mathbf{p}_0$ may be described in a parametric form:

$$\mathbf{p}(t) = \mathbf{p}_0 + t\,\Delta\mathbf{p} \tag{3.8}$$

where t is a parameter and $\Delta\mathbf{p} = [\Delta\mathbf{p}_u\ \Delta\mathbf{p}_v\ \Delta\mathbf{p}_w]^{\mathrm{T}}$ is the orientation vector of the line. Given a fixed camera position $\mathbf{c}$, a mapping $t \in \mathbb{R}^0 \rightarrow \mathbf{r} \in \mathbb{R}^3$ defines a parameterized space curve $\mathbf{r}(t)$ lying on the mirror surface, which describes the position of the reflection point as t varies (see Figure 3.2b). Consequently, through a perspective projection, $\mathbf{r}(t)$ is mapped to another parameterized curve $\mathbf{q}(t)$ on the image plane. We denote the first-order derivatives (tangent vector) of $\mathbf{r}(t)$ and $\mathbf{q}(t)$, respectively by $\dot{\mathbf{r}}$ and $\dot{\mathbf{q}}$, and denote their second-order derivatives respectively by $\ddot{\mathbf{r}}$ and $\ddot{\mathbf{q}}$. When $t = t_0 = 0$, we denote $\mathbf{r}(t_0)$ by $\mathbf{r}_0$, which is the reflection point of $\mathbf{p}_0$ on the mirror surface and can be set as the origin of the principal reference system. Accordingly, the values of $\dot{\mathbf{r}}$, $\dot{\mathbf{q}}$, $\ddot{\mathbf{r}}$ and $\ddot{\mathbf{q}}$ evaluated at t_0 are denoted by $\dot{\mathbf{r}}_0$, $\dot{\mathbf{q}}_0$, $\ddot{\mathbf{r}}_0$ and $\ddot{\mathbf{q}}_0$. Throughout this chapter, if there is no further explanation, we always assume that we evaluate $\dot{\mathbf{r}}$, $\dot{\mathbf{q}}$, $\ddot{\mathbf{r}}$ and $\ddot{\mathbf{q}}$ at t_0, and omit the subscript 0 to make the notation easier on the eye. Furthermore, we denote the curvature of the image plane curve $\mathbf{q}(t)$ as $\kappa^q(t)$, which is evaluated at t_0 as κ^q.

Our approach is to perform differential analysis around the point of intersection $\mathbf{r}_0$ of two or more scene lines. Specifically, we derive analytical expressions for the first- and second-order derivatives of deformed image curves at $\mathbf{q}_0$, in terms of surface parameters up to the third-order (s, a, b, c, e, f, g, h) (see Section 3.3). By comparing these analytical formulas with their corresponding local measurements in the image, we impose a set of constraints on the unknown surface parameters. The resulting constraint system allows us to recover the mirror surface locally around the reflection point $\mathbf{r}_0$ up to the third-order accuracy (see Section 3.5).

3.3 Direct Problem

In this section, we study the direct problem, that is, how the geometry of a specular surface determines its reflective behavior on a pattern line $\mathbf{p}(t)$ through a point $\mathbf{p}_0$, and consequently affects image plane measurements of $\mathbf{q}(t)$. Specifically, we carry out a differential analysis on the fundamental relationship between the image measurements such as $\dot{\mathbf{q}}_0$, $\ddot{\mathbf{q}}_0$, κ^q and the first-, second- and third-order surface parameters of the mirror surface around the reflection point $\mathbf{r}_0$.

The main results of this analysis are summarized in Table 3.2: i) first-order derivative of the image curve $\mathbf{q}(t)$ at $\mathbf{q}_0$ (namely, $\dot{\mathbf{q}}_0$) is a function of the first- and second-order parameters of the surface at $\mathbf{r}_0$ (namely,

distance s, tangent plane orientation and curvature parameters a, b, c); and ii) second-order derivative of the image curve $\mathbf{q}(t)$ at $\mathbf{q}_0$ (namely, $\ddot{\mathbf{q}}_0$) is a function of the first-, second- and third-order parameters of the surface at $\mathbf{r}_0$ (namely, s, a, b, c, e, f, g, h); in particular, $\ddot{\mathbf{q}}_0$ depends linearly on e, f, g, h.

Image Derivatives at $\mathbf{q}_0$	**Local Surface Parameters at $\mathbf{r}_0$**	**Equations**
First-order derivative of $\mathbf{q}(t)$ (i.e., $\dot{\mathbf{q}}_0$)	distance s, tangent plane, curvature (i.e., a, b, c)	Eq. (3.9) Eq. (3.28)
Second-order derivative of $\mathbf{q}(t)$ (i.e., $\ddot{\mathbf{q}}_0$)	distance s, tangent plane, curvature (i.e., a, b, c), third-order parameters (i.e., e, f, g, h)	Eq. (3.37) Eq. (3.50)

Table 3.2: Dependency between image measurements and local surface parameters. The last column points to the most meaningful equations

In Section 3.3.1, we study the first-order differential behavior of $\mathbf{r}(t)$ around t_0 . Proposition 2 expresses the relationship among $\dot{\mathbf{r}}_0$ (i.e., the first-order derivative of the reflected curve $\mathbf{r}(t)$ at $\mathbf{r}_0$), the surface shape up to second-order (i.e., s, a, b, c) and the scene line geometry. Proposition 4 relates $\dot{\mathbf{r}}_0$ to $\dot{\mathbf{q}}_0$. By combining these two propositions we demonstrate the first row of Table 3.2.

Then, in Section 3.3.2, we study the second-order differential behavior of $\mathbf{r}(t)$ around t_0. Proposition 5 expresses the relationship among $\ddot{\mathbf{r}}_0$ (i.e., the second-order derivative of the reflected curve $\mathbf{r}(t)$ at $\mathbf{r}_0$), surface shape up to third-order (i.e., s, a, b, c, e, f, g, h) and the scene line geometry. Proposition 6 relates $\ddot{\mathbf{r}}_0$ to $\ddot{\mathbf{q}}_0$. By combining these two propositions we demonstrate the second row of Table 3.2.

3.3.1 First Order Analysis

3.3.1.1 First-order Derivative of $\mathbf{r}(t)$

In this section we study the first-order differential behavior of $\mathbf{r}(t)$ around t_0.

Proposition 2 The first-order derivative of the surface curve $\mathbf{r}(t)$ at t_0 (i.e., $\dot{\mathbf{r}}_0$; namely, the tangent vector of $\mathbf{r}(t)$ at $\mathbf{r}_0$) is a function of the first-order surface parameter s and second-order surface parameters (a, b, c). Specifically, $\dot{\mathbf{r}}_0$ may be expressed as

$$\dot{\mathbf{r}}_0 = \begin{bmatrix} \dot{u} \\ \dot{v} \\ \dot{w} \end{bmatrix} = -\frac{1}{\Delta} \begin{bmatrix} J_v - 2b\cos\theta & 2c\cos\theta & 0 \\ 2c\cos\theta & J_u - 2a\cos\theta & 0 \\ 0 & 0 & \Delta \end{bmatrix} \begin{bmatrix} B_u \\ B_v \\ 0 \end{bmatrix}, \tag{3.9}$$

where

$$B_u = \frac{(\Delta\mathbf{p})_w \cos\theta \sin\theta - (\Delta\mathbf{p})_u \cos^2\theta}{||\,\mathbf{p}_0\,||}, \quad B_v = -\frac{(\Delta\mathbf{p})_v}{||\,\mathbf{p}_0\,||}, \tag{3.10}$$

$$J_v = \frac{s + ||\,\mathbf{p}_0\,||}{s\,||\,\mathbf{p}_0\,||} = \frac{1}{s} + \frac{1}{||\,\mathbf{p}_0\,||}, \quad J_u = J_v \cos^2\theta, \tag{3.11}$$

$$J_w = \frac{||\,\mathbf{p}_0\,|| - s}{s\,||\,\mathbf{p}_0\,||} \sin\theta\cos\theta \tag{3.12}$$

$$\Delta = (J_u - 2a\cos\theta)(J_v - 2b\cos\theta) - 4c^2\cos^2\theta. \tag{3.13}$$

Proof: The Fermat principle [8] stipulates that the reflection point $\mathbf{r}$ must be located on the mirror surface in such a way that locally minimizes the length of the specular path from the scene point $\mathbf{p}$ to the camera $\mathbf{c}$. Thus, the reflection point $\mathbf{r}$ can be found by solving a constrained optimization problem, that is:

$$\begin{aligned} &\text{minimize} && ||\,\mathbf{r} - \mathbf{p}\,|| + ||\,\mathbf{r} - \mathbf{c}\,|| \\ &\text{subject to} && g(\mathbf{r}) = 0, \end{aligned} \tag{3.14}$$

where $g(u, v, w) = 0$ denotes the implicit function of the mirror surface, which can be obtained from its Monge form (3.7) as:

$$\begin{aligned} g(u,v,w) = {} & w - \frac{1}{2!}(a\,u^2 + 2c\,uv + b\,v^2) - \\ & \frac{1}{3!}(e\,u^3 + 3f\,u^2v + 3g\,uv^2 + h\,v^3) + \cdots = 0. \end{aligned} \tag{3.15}$$

By applying the Lagrange Multiplier Theorem as demonstrated by Chen and Arvo [13], we obtain

$$\begin{cases} \mathbf{x}_r(\mathbf{r}) + \mathbf{x}_i(\mathbf{r}, t) + \lambda\nabla g(\mathbf{r}) = \mathbf{0} \\ g(\mathbf{r}) = 0 \end{cases}, \tag{3.16}$$

where we have used $\mathbf{x}_i$ and $\mathbf{x}_r$ to denote the incident direction (depending on both $\mathbf{r}$ and t) and reflection direction (depending on $\mathbf{r}$), which are defined respectively as

$$\mathbf{x}_i = \frac{\mathbf{r} - \mathbf{p}(t)}{||\,\mathbf{r} - \mathbf{p}(t)\,||}, \tag{3.17}$$

$$\mathbf{x}_r = \frac{\mathbf{r} - \mathbf{c}}{||\,\mathbf{r} - \mathbf{c}\,||}. \tag{3.18}$$

Notice that $\mathbf{x}_r$ has already been introduced as $\mathbf{d}$ in Eq. (3.1). Here we emphasize the dependence on $\mathbf{r}$. Eq. (3.16) is a non-linear system of four equations in five unknowns (t, $\mathbf{r}$ and λ), and obviously, $(0, \mathbf{r}_0, 2\cos\theta_0)$[2] is one of its solutions. We refer to it as *Fermat equation* and denote it by $\mathbf{F}(t, \mathbf{r}, \lambda) = \mathbf{0}$, where each component of $\mathbf{F} = [F_1\ F_2\ F_3\ F_4]$ captures the left-hand side of one of the equations in (3.16).

We take advantage of Implicit Function Theorem to study the differential behavior of $\mathbf{r}(t)$ around t_0. The theorem states that if $(t_0, \mathbf{r}_0, \lambda_0)$ is a solution of $\mathbf{F}(t, \mathbf{r}, \lambda) = 0$ and the Jacobian

$$\Delta \ = \ \det\left[\frac{\partial \mathbf{F}(t, \mathbf{r}, \lambda)}{\partial(\mathbf{r}, \lambda)}\right]\Bigg|_{t=t_0} \neq 0, \tag{3.19}$$

the smooth mapping function $\mathbf{r}(t)$ exists around the neighborhood of $\mathbf{r}_0$. Furthermore, we may compute the derivative of $\mathbf{r}(t)$ with respect to t by differentiating $\mathbf{F}$ with respect to t. By chain rule, we obtain:

$$\mathbf{B}(t, \mathbf{r}, \lambda) \ = \ -\mathbf{J}(t, \mathbf{r}, \lambda)\, \mathbf{S}(t), \tag{3.20}$$

where

$$\begin{aligned} \mathbf{B}(t, \mathbf{r}, \lambda) &= \frac{\partial \mathbf{F}(t, \mathbf{r}, \lambda)}{\partial t}, \quad \mathbf{J}(t, \mathbf{r}, \lambda) \ = \ \frac{\partial \mathbf{F}(t, \mathbf{r}, \lambda)}{\partial(\mathbf{r}, \lambda)}, \\ \mathbf{S}(t) &= [\dot{u}(t)\ \dot{v}(t)\ \dot{w}(t)\ \dot{\lambda}(t)]^T = [\dot{\mathbf{r}}(t)\ \dot{\lambda}(t)]^T. \end{aligned}$$

We may express **B** and **J** as follows:

$$\mathbf{B} \ = \ \begin{bmatrix} \mathbf{B}_{31} \\ 0 \end{bmatrix}, \quad \mathbf{J} \ = \ \begin{bmatrix} \mathbf{J}_{33} & \nabla g(\mathbf{r}) \\ (\nabla g(\mathbf{r}))^{\mathrm{T}} & 0 \end{bmatrix}, \tag{3.21}$$

where the sub-matrices $\mathbf{B}_{31}$ and $\mathbf{J}_{33}$ (the subscript denotes the matrix dimension) are defined as

$$\mathbf{B}_{31} \ = \ -\frac{(\mathbf{I} - \mathbf{x}_r \mathbf{x}_r{}^{\mathrm{T}})\Delta\mathbf{p}}{||\,\mathbf{r} - \mathbf{p}(t)\,||}, \quad \mathbf{J}_{33} \ = \ \frac{(\mathbf{I} - \mathbf{x}_r \mathbf{x}_r{}^{\mathrm{T}})}{||\,\mathbf{r} - \mathbf{p}(t)\,||} + \frac{(\mathbf{I} - \mathbf{x}_i \mathbf{x}_i{}^{\mathrm{T}})}{||\,\mathbf{r} - \mathbf{c}\,||} + \lambda \mathbf{H}_g. \tag{3.22}$$

[2] $\lambda_0 = 2\cos\theta_0$ is obtained by dot product Eq. (3.16) with $\nabla g(\mathbf{r}_0)$.

Here, g denotes the implicit definition of the surface and ∇g, $\mathbf{H}_g$ are respectively the gradient vector and the Hessian matrix of g. It follows from Eq. (3.15) that

$$\begin{aligned}
\nabla g &= \begin{bmatrix} -a\,u - c\,v - \frac{1}{2}e\,u^2 - f\,uv - \frac{1}{2}g\,v^2 + \cdots \\ -c\,u - b\,v - \frac{1}{2}f\,u^2 - g\,uv - \frac{1}{2}h\,v^2 + \cdots \\ 1 \end{bmatrix}, \\
\mathbf{H}_g &= \begin{bmatrix} -a - e\,u - f\,v + \cdots & -c - f\,u - g\,v + \cdots & 0 \\ -c - f\,u - g\,v + \cdots & -b - g\,u - h\,v + \cdots & 0 \\ 0 & 0 & 0 \end{bmatrix}. \qquad (3.23)
\end{aligned}$$

In order to compute $\dot{\mathbf{r}}(t)$ at $\mathbf{r}_0$, we need to evaluate $\mathbf{B}, \mathbf{J}$ at $t = t_0$ (and thus, at $\mathbf{p} = \mathbf{p}_0$, $\mathbf{r} = \mathbf{r}_0$ and $\lambda = \lambda_0 = 2\cos\theta_0$). At that end, we note that since we work in the principal reference system, $\mathbf{x}_r$ and $\mathbf{x}_i$ may be expressed in terms of the reflection angle θ:

$$\mathbf{x}_i = (\sin\theta, 0, -\cos\theta)^{\mathrm{T}}, \quad \mathbf{x}_r = (-\sin\theta, 0, -\cos\theta)^{\mathrm{T}}.$$

Thus, we may expand Eq. (3.20) as follows:

$$\begin{bmatrix} B_u \\ B_v \\ B_w \\ 0 \end{bmatrix} = -\begin{bmatrix} J_u - 2a\cos\theta & -2c\cos\theta & J_w & 0 \\ -2c\cos\theta & J_v - 2b\cos\theta & 0 & 0 \\ J_w & 0 & J_v\sin^2\theta & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} \dot{u}(0) \\ \dot{v}(0) \\ \dot{w}(0) \\ \dot{\lambda}(0) \end{bmatrix}, \qquad (3.24)$$

where $\cos\theta$ is given by Eq. (3.6), $\sin^2\theta = 1 - \cos^2\theta$ and B_u, B_v, B_w, J_u and J_v are defined above. It follows from the last row of Eq. (3.24) that $\dot{w}(0) = 0$. Consequently,

$$\begin{bmatrix} B_u \\ B_v \\ B_w \end{bmatrix} = -\begin{bmatrix} J_u - 2a\cos\theta & -2c\cos\theta & 0 \\ -2c\cos\theta & J_v - 2b\cos\theta & 0 \\ J_w & 0 & 1 \end{bmatrix} \begin{bmatrix} \dot{u}(0) \\ \dot{v}(0) \\ \dot{\lambda}(0) \end{bmatrix}. \qquad (3.25)$$

By inverting the matrix in Eq. (3.25), we obtain the first-order derivative of the surface curve $\mathbf{r}(t)$ at $\mathbf{r}_0$ (i.e., t_0) and the proposition is proved. The determinant of the matrix in Eq. (3.25) ($\Delta = (J_u - 2a\cos\theta)(J_v - 2b\cos\theta) - 4c^2\cos^2\theta$) is equal to the Jacobian in Eq. (3.19). ■

Proposition 2 tells us that $\dot{\mathbf{r}}_0$ is *not* a function of the third or higher order parameters of the surface around the

reflection point. A similar relationship was derived by Zisserman *et al.* [84] in the dual context of a moving observer. Note that Proposition 2 holds under the condition of $\Delta \neq 0$ in Eq. (3.19), which guarantees that the mapping $\mathbf{r}(t)$ is unique in the neighborhood of $(t_0, \mathbf{r}_0, \lambda_0)$. From now on, we shall always assume that $\Delta \neq 0$. This assumption is reasonable as we shall see in more detail in Section 3.4.1.

If we call ϕ the angle between $\dot{\mathbf{r}}$ and $\mathbf{u}$ axis, the orientation of the tangent vector of $\mathbf{r}(t)$ at $\mathbf{r}_0$ within the tangent plane may be expressed as

$$\tan\phi = \frac{\dot{v}}{\dot{u}} = \frac{(J_u - 2a\cos\theta)B_v + 2cB_u\cos\theta}{(J_v - 2b\cos\theta)B_u + 2cB_v\cos\theta}. \tag{3.26}$$

3.3.1.2 Relationship Between $\dot{\mathbf{r}}$ and $\dot{\mathbf{q}}$

Proposition 3 The direction of the tangent vector $\dot{\mathbf{r}}$ is directly related to the direction of the tangent vector $\dot{\mathbf{q}}$ by the following relationship (up to a sign)

$$\frac{\dot{\mathbf{r}}}{||\,\dot{\mathbf{r}}\,||} \;=\; \frac{\mathbf{n}_r \times (\mathbf{c}/\,||\,\mathbf{c}\,|| \times \dot{\mathbf{q}}/\,||\,\dot{\mathbf{q}}\,||)}{||\,\mathbf{n}_r \times (\mathbf{c}/\,||\,\mathbf{c}\,|| \times \dot{\mathbf{q}}/\,||\,\dot{\mathbf{q}}\,||)\,||}. \tag{3.27}$$

Proof: It is not difficult to show that $\dot{\mathbf{q}}$ is proportional to the perspective projection of $\dot{\mathbf{r}}$ into the image plane. Thus, the plane defined by $\mathbf{c}$ and $\dot{\mathbf{q}}$ contains $\dot{\mathbf{r}}$. Let $\mathbf{n}_t$ denote the normal vector of this plane, then we have $\langle \mathbf{n}_t, \dot{\mathbf{r}} \rangle = 0$. Since $\langle \mathbf{n}_r, \dot{\mathbf{r}} \rangle = 0$, we have $\frac{\dot{\mathbf{r}}}{\|\dot{\mathbf{r}}\|} = \frac{\mathbf{n}_t \times \mathbf{n}_r}{\|\mathbf{n}_t \times \mathbf{n}_r\|}$ up to a sign, which gives rise to Eq. (3.27). ■

More generally,

Proposition 4 Let $\mathbf{r}(t)$ be the surface curve and $\mathbf{q}(t)$ be its camera image. Then, respective tangent vectors are linearly related:

$$\dot{\mathbf{q}} = \mathbf{T}\,\dot{\mathbf{r}} \tag{3.28}$$

where $\mathbf{T}$ is a 3×3 matrix function of the camera parameters, $\mathbf{r}_0$ and $\mathbf{q}_0$. Specifically, the matrix $\mathbf{T}$ is defined as follows:

$$\mathbf{T} \;=\; \frac{l}{s\,\langle \mathbf{d}, \mathbf{v} \rangle}\left[\mathbf{I} - \frac{\mathbf{d}\mathbf{v}^{\mathrm{T}}}{\langle \mathbf{d}, \mathbf{v} \rangle}\right] \tag{3.29}$$

where $\mathbf{d} = (\mathbf{q} - \mathbf{c})/\,||\,\mathbf{q} - \mathbf{c}\,||$, $s = ||\,\mathbf{r} - \mathbf{c}\,||$, $\mathbf{v}$ is the view direction and l is the distance from $\mathbf{c}$ of the image plane along $\mathbf{v}$.

Proof: Let $\gamma(t) = || \mathbf{q}(t) - \mathbf{c} || / || \mathbf{r}(t) - \mathbf{c} ||$ be the ratio between the distance from $\mathbf{c}$ to $\mathbf{q}(t)$ and from $\mathbf{c}$ to $\mathbf{r}(t)$. We may express the image plane curve $\mathbf{q}(t)$ as follows:

$$\mathbf{q}(t) - \mathbf{c} \;=\; \gamma(t)(\mathbf{r}(t) - \mathbf{c}). \tag{3.30}$$

Thus, $\mathbf{q}(t)$ satisfies

$$\langle \mathbf{q}(t) - \mathbf{c}, \mathbf{v} \rangle \;=\; l,$$

which can be expressed in terms of γ using Eq. (3.30), yielding

$$\gamma(t) \langle \mathbf{r}(t) - \mathbf{c}, \mathbf{v} \rangle \;=\; l. \tag{3.31}$$

Differentiating Eq. (3.31) with respect to t, we get

$$\dot{\gamma} \langle \mathbf{r} - \mathbf{c}, \mathbf{v} \rangle + \gamma \langle \dot{\mathbf{r}}, \mathbf{v} \rangle \;=\; 0. \tag{3.32}$$

We may then solve for γ from Eq. (3.31) and Eq. (3.1) and $\dot{\gamma}$ from Eq. (3.32), obtaining

$$\gamma \;=\; \frac{l}{\langle \mathbf{r}(t) - \mathbf{c}, \mathbf{v} \rangle} \;=\; \frac{l}{s \langle \mathbf{d}, \mathbf{v} \rangle}, \tag{3.33}$$

$$\dot{\gamma} \;=\; -\frac{l \langle \dot{\mathbf{r}}, \mathbf{v} \rangle}{s^2 \langle \mathbf{d}, \mathbf{v} \rangle^2}. \tag{3.34}$$

Note that γ and $\dot{\gamma}$ are both evaluated at $t = t_0$. Using Eq. (3.33) and Eq. (3.34), we can differentiate Eq. (3.30) with respect to t and compute $\dot{\mathbf{q}}$ as follows:

$$\dot{\mathbf{q}} \;=\; \gamma \dot{\mathbf{r}} + s \dot{\gamma} \mathbf{d}, \tag{3.35}$$

$$\;=\; \frac{l}{s \langle \mathbf{d}, \mathbf{v} \rangle} \left[\mathbf{I} - \frac{\mathbf{d}\mathbf{v}^{\mathrm{T}}}{\langle \mathbf{d}, \mathbf{v} \rangle} \right] \dot{\mathbf{r}} \;=\; \mathbf{T}\, \dot{\mathbf{r}}, \tag{3.36}$$

■

3.3.2 Second Order Analysis

3.3.2.1 Second-Order Derivative of $\mathbf{r}(t)$

In this section we study the second-order differential behavior of $\mathbf{r}(t)$ around $t = t_0$.

Proposition 5 The second-order derivative of the surface curve $\mathbf{r}(t)$ at $\mathbf{r}_0$ (i.e., t_0) is a function of the first-order surface parameter s, second-order surface parameters a, b, c, and third-order surface parameters e, f, g, h. Particularly, it depends linearly on the third-order surface parameters e, f, g, h. Specifically, $\ddot{\mathbf{r}}$ may be expressed as

$$\ddot{\mathbf{r}} = \ddot{\mathbf{r}}_1 + \ddot{\mathbf{r}}_2 = \begin{bmatrix} \ddot{u}_1 \\ \ddot{v}_1 \\ \ddot{w}_1 \end{bmatrix} + \begin{bmatrix} \ddot{u}_2 \\ \ddot{v}_2 \\ 0 \end{bmatrix} \tag{3.37}$$

where the first term is

$$\begin{bmatrix} \ddot{u}_1 \\ \ddot{v}_1 \end{bmatrix} = -\frac{1}{\Delta} \begin{bmatrix} J_v - 2b\cos\theta & 2c\cos\theta \\ 2c\cos\theta & J_u - 2a\cos\theta \end{bmatrix} \begin{bmatrix} D_1 + J_w\ddot{w} \\ D_2 \end{bmatrix} \tag{3.38}$$

$$\ddot{w}_1 = -a\dot{u}^2 - 2c\dot{u}\dot{v} - b\dot{v}^2, \tag{3.39}$$

and D_1, D_2 (which are both functions of $\dot{u}$ and $\dot{v}$) are derived in the proof. The second term is

$$\begin{bmatrix} \ddot{u}_2 \\ \ddot{v}_2 \end{bmatrix} = \frac{2\cos\theta}{\Delta} \begin{bmatrix} J_v - 2b\cos\theta & 2c\cos\theta \\ 2c\cos\theta & J_u - 2a\cos\theta \end{bmatrix} \times \begin{bmatrix} \dot{u}^2 & 2\dot{u}\dot{v} & \dot{v}^2 & 0 \\ 0 & \dot{u}^2 & 2\dot{u}\dot{v} & \dot{v}^2 \end{bmatrix} \begin{bmatrix} e \\ f \\ g \\ h \end{bmatrix}. \tag{3.40}$$

In both terms, the quantities J_u, J_v, Δ are defined in Proposition 2.

Proof: By differentiating Eq. (3.20) with respect to t, we have

$$\frac{d\mathbf{B}(t, \mathbf{r}, \lambda)}{dt} = -\frac{d\mathbf{J}(t, \mathbf{r}, \lambda)}{dt}\,\mathbf{S} - \mathbf{J}\,\frac{d\mathbf{S}(t)}{dt}, \tag{3.41}$$

where

$$\frac{d\mathbf{S}}{dt} \;=\; [\ddot{\mathbf{r}}(t)\;\; \ddot{\lambda}(t)]^{\mathrm{T}} \;=\; [\ddot{u}(t)\;\; \ddot{v}(t)\;\; \ddot{w}(t)\;\; \ddot{\lambda}(t)]^{\mathrm{T}}$$

Our goal is to compute $\ddot{\mathbf{r}}$. For this purpose, we are required to compute the components $d\mathbf{B}/dt$ and $d\mathbf{J}/dt$ in Eq. (3.41). Note that we should think of $(\mathbf{r}, \lambda)$ as a function of t (i.e., $\mathbf{r}(t)$ which is guaranteed to exist from our first-order analysis) while calculating these derivatives. This is indicated by notation d/dt instead of $\partial/\partial t$. These expressions are derived for a general t, and will be evaluated at the specular path through $\mathbf{p}_0$, that is, $t = 0$. In this derivation, we will use the identities:

$$\mathbf{N}_r \;=\; \mathbf{I} - \mathbf{x}_r\mathbf{x}_r{}^{\mathrm{T}} \;=\; \begin{bmatrix} \cos^2\theta & 0 & -\sin\theta\cos\theta \\ 0 & 1 & 0 \\ -\sin\theta\cos\theta & 0 & \sin^2\theta \end{bmatrix},$$

$$\mathbf{N}_i \;=\; \mathbf{I} - \mathbf{x}_i\mathbf{x}_i{}^{\mathrm{T}} \;=\; \begin{bmatrix} \cos^2\theta & 0 & \sin\theta\cos\theta \\ 0 & 1 & 0 \\ \sin\theta\cos\theta & 0 & \sin^2\theta \end{bmatrix}$$

$$\begin{array}{llll} \frac{\partial||\,\mathbf{r}-\mathbf{p}(t)\,||}{\partial t} &=& -\mathbf{x}_r{}^{\mathrm{T}}\Delta\mathbf{p}, & \frac{\partial\mathbf{x}_r}{\partial t} \;=\; -\frac{\mathbf{N}_r}{||\,\mathbf{r}-\mathbf{p}(t)\,||}\Delta\mathbf{p}, \\ \frac{\partial||\,\mathbf{r}-\mathbf{p}(t)\,||}{\partial\mathbf{r}} &=& \mathbf{x}_r{}^{\mathrm{T}}, & \frac{\partial\mathbf{x}_r}{\partial\mathbf{r}} \;=\; \frac{\mathbf{N}_r}{||\,\mathbf{r}-\mathbf{p}(t)\,||}, \\ \frac{\partial\mathbf{N}_r}{\partial t} &=& \frac{\mathbf{N}_r\Delta\mathbf{p}\mathbf{x}_r{}^{\mathrm{T}}+\mathbf{x}_r\Delta\mathbf{p}^{\mathrm{T}}\mathbf{N}_r}{||\,\mathbf{r}-\mathbf{p}(t)\,||}, & \frac{\partial(\mathbf{N}_r\Delta\mathbf{p})}{\partial\mathbf{r}} \;=\; -\frac{\mathbf{N}_r\mathbf{x}_r{}^{\mathrm{T}}\Delta\mathbf{p}+\mathbf{x}_r\Delta\mathbf{p}^{\mathrm{T}}\mathbf{N}_r}{||\,\mathbf{r}-\mathbf{p}(t)\,||} \end{array}$$

We start by differentiating **B** and obtain:

$$\frac{\partial\mathbf{B}_{31}}{\partial t} \;=\; -\mathbf{A}\Delta\mathbf{p}, \quad \frac{\partial\mathbf{B}_{31}}{\partial\mathbf{r}} \;=\; \mathbf{A},$$

where the matrix factor **A** is defined as

$$\mathbf{A} \;=\; \frac{\mathbf{N}_r\mathbf{x}_r{}^{\mathrm{T}}\Delta\mathbf{p} + \mathbf{N}_r\Delta\mathbf{p}\mathbf{x}_r{}^{\mathrm{T}} + \mathbf{x}_r\Delta\mathbf{p}^{\mathrm{T}}\mathbf{N}_r}{||\,\mathbf{r}-\mathbf{p}(t)\,||^2}$$

Consequently, we get from the chain rule

$$\begin{aligned}\frac{d\mathbf{B}(t,\mathbf{r},\lambda)}{dt} &= \frac{\partial \mathbf{B}}{\partial t}+\frac{\partial \mathbf{B}}{\partial(\mathbf{r},\lambda)}\dot{\mathbf{S}} = \begin{bmatrix} -\mathbf{A}\Delta\mathbf{p} \\ 0 \end{bmatrix}+\begin{bmatrix} \mathbf{A} & 0 \\ 0 & 0 \end{bmatrix}\begin{bmatrix} \dot{\mathbf{r}}(t) \\ \dot{\lambda}(t) \end{bmatrix} \\ &= \begin{bmatrix} \mathbf{A}(\dot{\mathbf{r}}(t)-\Delta\mathbf{p}) \\ 0 \end{bmatrix}. \end{aligned} \tag{3.42}$$

Similarly, we may compute $d\mathbf{J}/dt$ through $\partial\mathbf{J}/\partial t$ and $\partial\mathbf{J}/\partial(\mathbf{r},\lambda)$ as follows:

$$\begin{aligned}\frac{\partial \mathbf{J}}{\partial t} &= \begin{bmatrix} \frac{\partial \mathbf{J}_{33}}{\partial t} & 0 \\ 0 & 0 \end{bmatrix} = \begin{bmatrix} \frac{\partial \mathbf{N}_r/\,||\,\mathbf{r}-\mathbf{p}(t)\,||}{\partial t} & 0 \\ 0 & 0 \end{bmatrix} \\ &= \begin{bmatrix} \mathbf{A} & 0 \\ 0 & 0 \end{bmatrix}. \end{aligned}$$

The computation of $\partial\mathbf{J}/\partial(\mathbf{r},\lambda)$ becomes a little more complicated, since the result will be a third-order tensor, which, after being multiplied by $\mathbf{S}$, returns a matrix. This third-order tensor can be computed by calculating the differential matrix associated with each column of $\mathbf{J}$. Let

$$\mathbf{e}_1 = (1,0,0)^{\mathrm{T}},\quad \mathbf{e}_2 = (0,1,0)^{\mathrm{T}},\quad \mathbf{e}_3 = (0,0,1)^{\mathrm{T}}.$$

and define $\mathbf{V}^j$ as the jth column of a matrix $\mathbf{V}$. Using Eq. (3.22), we have

$$\mathbf{J}_{33}^j = \frac{\mathbf{N}_r^j}{||\,\mathbf{r}-\mathbf{p}(t)\,||}+\frac{\mathbf{N}_i^j}{||\,\mathbf{r}-\mathbf{c}\,||}+\lambda\mathbf{H}_g^j$$

Differentiating the first two components of $\mathbf{J}_{33}^j$, we have

$$\begin{aligned}\mathbf{C}_r^j &= \frac{\partial(\mathbf{N}_r^j)/\,||\,\mathbf{r}-\mathbf{p}(t)\,||}{\partial\mathbf{r}} = -\frac{\mathbf{N}_r\mathbf{x}_r{}^{\mathrm{T}}\mathbf{e}_j+\mathbf{N}_r\mathbf{e}_j\mathbf{x}_r{}^{\mathrm{T}}+\mathbf{x}_r\mathbf{e}_j{}^{\mathrm{T}}\mathbf{N}_r}{||\,\mathbf{r}-\mathbf{p}(t)\,||^2}, \\ \mathbf{C}_i^j &= \frac{\partial(\mathbf{N}_i^j)/\,||\,\mathbf{r}-\mathbf{c}\,||}{\partial\mathbf{r}} = -\frac{\mathbf{N}_i\mathbf{x}_i{}^{\mathrm{T}}\mathbf{e}_j+\mathbf{N}_i\mathbf{e}_j\mathbf{x}_i{}^{\mathrm{T}}+\mathbf{x}_i\mathbf{e}_j{}^{\mathrm{T}}\mathbf{N}_i}{||\,\mathbf{r}-\mathbf{c}\,||^2}. \end{aligned}$$

Furthermore, it follows from Eq. (3.23) that

$$\begin{aligned}
\mathbf{C}_h^1 &= \frac{\partial \mathbf{H}_g^1}{\partial \mathbf{r}} = \begin{bmatrix} -e & -f & 0 \\ -f & -g & 0 \\ 0 & 0 & 0 \end{bmatrix}, \quad \mathbf{C}_h^2 = \frac{\partial \mathbf{H}_g^2}{\partial \mathbf{r}} = \begin{bmatrix} -f & -g & 0 \\ -g & -h & 0 \\ 0 & 0 & 0 \end{bmatrix}, \\
\mathbf{C}_h^3 &= \frac{\partial \mathbf{H}_g^3}{\partial \mathbf{r}} = \mathbf{0}.
\end{aligned} \tag{3.43}$$

Consequently, we can compute

$$\frac{\partial \mathbf{J}_{33}^j}{\partial(\mathbf{r}, \lambda)} = \left[\mathbf{C}_r^j + \mathbf{C}_i^j + \mathbf{C}_h^j \;\; \mathbf{H}_g^j\right]_{34}.$$

Moreover, we have

$$\frac{\partial(\nabla g)}{\partial(\mathbf{r}, \lambda)} = \begin{bmatrix} \mathbf{H}_g & 0 \end{bmatrix}_{34}.$$

Define

$$\begin{aligned}
\mathbf{C} &= \begin{bmatrix} (\mathbf{C}_r^1 + \mathbf{C}_i^1)\dot{\mathbf{r}}(t) + \mathbf{H}_g^1\dot{\lambda}(t) & (\mathbf{C}_r^2 + \mathbf{C}_i^2)\dot{\mathbf{r}}(t) + \mathbf{H}_g^2\dot{\lambda}(t) & (\mathbf{C}_r^3 + \mathbf{C}_i^3)\dot{\mathbf{r}}(t) + \mathbf{H}_g^3\dot{\lambda}(t) \end{bmatrix}_{33}, \\
\mathbf{C}' &= \lambda \begin{bmatrix} \mathbf{C}_h^1\dot{\mathbf{r}}(t) & \mathbf{C}_h^2\dot{\mathbf{r}}(t) & \mathbf{C}_h^3\dot{\mathbf{r}}(t) \end{bmatrix} \\
&= \lambda \begin{bmatrix} -e\dot{u}(0) - f\dot{v}(0) & -f\dot{u}(0) - g\dot{v}(0) & 0 \\ -f\dot{u}(0) - g\dot{v}(0) & -g\dot{u}(0) - h\dot{v}(0) & 0 \\ 0 & 0 & 0 \end{bmatrix}.
\end{aligned}$$

It follows from the chain rule that

$$\begin{aligned}
\frac{d\mathbf{J}(t, \mathbf{r}, \lambda)}{dt} &= \begin{bmatrix} \mathbf{A} & 0 \\ 0 & 0 \end{bmatrix} + \begin{bmatrix} \mathbf{C} + \mathbf{C}' & \mathbf{H}_g\dot{\mathbf{r}}(t) \\ (\mathbf{H}_g\dot{\mathbf{r}}(t))^{\mathrm{T}} & 0 \end{bmatrix} \\
&= \begin{bmatrix} \mathbf{A} + \mathbf{C} + \mathbf{C}' & \mathbf{H}_g\dot{\mathbf{r}}(t) \\ (\mathbf{H}_g\dot{\mathbf{r}}(t))^{\mathrm{T}} & 0 \end{bmatrix}
\end{aligned} \tag{3.44}$$

Finally, by substituting equations (3.42) and (3.44) into Eq. (3.41), we obtain

$$\begin{bmatrix} \mathbf{A}(\dot{\mathbf{r}}(t) - \Delta\mathbf{p}) \\ 0 \end{bmatrix} = -\begin{bmatrix} \mathbf{A} + \mathbf{C} + \mathbf{C}' & \mathbf{H}_g\dot{\mathbf{r}}(t) \\ (\mathbf{H}_g\dot{\mathbf{r}}(t))^{\mathrm{T}} & 0 \end{bmatrix} \begin{bmatrix} \dot{\mathbf{r}}(t) \\ \dot{\lambda}(t) \end{bmatrix} - \begin{bmatrix} \mathbf{J}_{33} & \nabla g(\mathbf{r}) \\ (\nabla g(\mathbf{r}))^{\mathrm{T}} & 0 \end{bmatrix} \begin{bmatrix} \ddot{\mathbf{r}}(t) \\ \ddot{\lambda}(t) \end{bmatrix},$$

which may be expressed as

$$\mathbf{D} \;=\; -\begin{bmatrix} \mathbf{J}_{33} & \nabla g(\mathbf{r}) \\ (\nabla g(\mathbf{r}))^{\mathrm{T}} & 0 \end{bmatrix} \begin{bmatrix} \ddot{\mathbf{r}}(t) \\ \ddot{\lambda}(t) \end{bmatrix}, \tag{3.45}$$

where $\mathbf{D}$ is defined as

$$\begin{aligned} \mathbf{D} &= \begin{bmatrix} \mathbf{A}(\dot{\mathbf{r}}(t) - \Delta\mathbf{p}) \\ 0 \end{bmatrix} + \begin{bmatrix} \mathbf{A} + \mathbf{C} + \mathbf{C}' & \mathbf{H}_g\dot{\mathbf{r}}(t) \\ (\mathbf{H}_g\dot{\mathbf{r}}(t))^{\mathrm{T}} & 0 \end{bmatrix} \begin{bmatrix} \dot{\mathbf{r}}(t) \\ \dot{\lambda}(t) \end{bmatrix} \\ &= \begin{bmatrix} \mathbf{A}(\dot{\mathbf{r}}(t) - \Delta\mathbf{p}) + (\mathbf{A} + \mathbf{C} + \dot{\lambda}(t)\mathbf{H}_g)\dot{\mathbf{r}}(t) \\ \\ (\mathbf{H}_g\dot{\mathbf{r}}(t))^{\mathrm{T}}\dot{\mathbf{r}}(t) \end{bmatrix} + \begin{bmatrix} \mathbf{C}'\dot{\mathbf{r}}(t) \\ 0 \end{bmatrix} \\ &= \mathbf{D}_1 + \mathbf{D}_2. \end{aligned} \tag{3.46}$$

Notice that the third-order surface parameters e, f, g, h (of the Monge form; see Eq. (3.7)) only appear in the second term $\mathbf{D}_2$. From the last row of Eq. (3.45), we obtain an expression in the second-order parameters a, b, c of the Monge form:

$$a\dot{u}^2(t) + 2c\dot{u}(t)\dot{v}(t) + b\dot{v}^2(t) - \ddot{w}(t) \;=\; 0,$$

which yields

$$\ddot{w}(0) \;=\; a\dot{u}^2(0) + 2c\dot{u}(0)\dot{v}(0) + b\dot{v}^2(0). \tag{3.47}$$

In addition, we can solve for $\ddot{u}(0)$ and $\ddot{v}(0)$ from the first two rows of Eq. (3.45). Specifically,

$$\begin{bmatrix} D_1 + D_1' + J_w\ddot{w}(0) \\ D_2 + D_2' \end{bmatrix} \;=\; -\begin{bmatrix} J_u - 2a\cos\theta & -2c\cos\theta \\ -2c\cos\theta & J_v - 2b\cos\theta \end{bmatrix} \begin{bmatrix} \ddot{u}(0) \\ \ddot{v}(0) \end{bmatrix},$$

where D_1, D_2 and D_1', D_2' denote the first two rows of $\mathbf{D}_1$ and $\mathbf{D}_2$ respectively. In particular, it follows from Eq. (3.46) that

$$\begin{bmatrix} D_1 & D_2 & D_3 \end{bmatrix}^T \;=\; (2\mathbf{A} + \mathbf{C} + \dot{\lambda}(t)\mathbf{H}_g)\dot{\mathbf{r}}(t) - \mathbf{A}\Delta\mathbf{p}, \tag{3.48}$$

and

$$\begin{bmatrix} D_1' \\ D_2' \end{bmatrix} = -2\cos\theta \begin{bmatrix} \dot{u}^2 & 2\dot{u}\dot{v} & \dot{v}^2 & 0 \\ 0 & \dot{u}^2 & 2\dot{u}\dot{v} & \dot{v}^2 \end{bmatrix} \begin{bmatrix} e \\ f \\ g \\ h \end{bmatrix}. \tag{3.49}$$

Consequently,

$$\begin{bmatrix} \ddot{u}(0) \\ \ddot{v}(0) \end{bmatrix} = -\frac{1}{\Delta} \begin{bmatrix} J_v - 2b\cos\theta & 2c\cos\theta \\ 2c\cos\theta & J_u - 2a\cos\theta \end{bmatrix} \begin{bmatrix} D_1 + D_1' + J_w\ddot{w}(0) \\ D_2 + D_2' \end{bmatrix}.$$

Hence, by combining with Eq. (3.47), the proposition is proven. ■

Note that Eq. (3.39) expresses Meusnier's Theorem (see [12]). Additionally, note that the second term of Eq. (3.37) depends linearly on the third-order surface parameters. Finally, when the third-order terms of the Monge form of the mirror surface are equal to zero, $\mathbf{C}_h^j = 0$ for $j = 1, 2, 3$ and the second term of Eq. (3.37) is zero. This leads to a special case in surface reconstruction as we shall see in Section 3.5.1.

3.3.2.2 Relationship Between $\dot{\mathbf{r}}$, $\ddot{\mathbf{r}}$ and $\ddot{\mathbf{q}}$

Proposition 6 Let $\mathbf{r}(t)$ be the surface curve and $\mathbf{q}(t)$ its camera image. Then, $\ddot{\mathbf{r}}$ and $\ddot{\mathbf{q}}$ are linearly related:

$$\ddot{\mathbf{q}} = \mathbf{T}\left[\ddot{\mathbf{r}} - \alpha\dot{\mathbf{r}}\right], \tag{3.50}$$

where the 3×3 matrix $\mathbf{T}$ and the scalar α are functions of the camera parameters, $\mathbf{r}_0$, $\mathbf{q}_0$ and $\dot{\mathbf{r}}$. Specifically, $\mathbf{T}$ is defined in Eq. (3.29) and α as follows:

$$\alpha = \frac{2}{s\left\langle \mathbf{d}, \mathbf{v} \right\rangle} \left\langle \dot{\mathbf{r}}, \mathbf{v} \right\rangle. \tag{3.51}$$

Proof: To relate the second-order derivative of $\mathbf{r}(t)$ to that of its image plane projection $\mathbf{q}(t)$, we differentiate Eq. (3.35) with respect to t, obtaining

$$\ddot{\mathbf{q}} = \gamma\,\ddot{\mathbf{r}} + 2\dot{\gamma}\,\dot{\mathbf{r}} + \ddot{\gamma}\,s\,\mathbf{d} \tag{3.52}$$

where $\gamma, \dot{\gamma}$ are defined in equations (3.33) and (3.34), and $\ddot{\gamma}$ may be computed from further differentiating Eq. (3.32), yielding

$$\ddot{\gamma} \;=\; -\frac{2\dot{\gamma}\left\langle \dot{\mathbf{r}}, \mathbf{v}\right\rangle + \gamma\left\langle \ddot{\mathbf{r}}, \mathbf{v}\right\rangle}{s\left\langle \mathbf{d}, \mathbf{v}\right\rangle} \tag{3.53}$$

Thus, we get an analytical expression for $\ddot{\mathbf{q}}$ by substituting into Eq. (3.52), that is

$$\begin{aligned} \ddot{\mathbf{q}} &= \gamma\left(\ddot{\mathbf{r}} - \frac{\langle \ddot{\mathbf{r}}, \mathbf{v}\rangle}{\langle \mathbf{d}, \mathbf{v}\rangle}\mathbf{d}\right) + 2\dot{\gamma}\left(\dot{\mathbf{r}} - \frac{\langle \dot{\mathbf{r}}, \mathbf{v}\rangle}{\langle \mathbf{d}, \mathbf{v}\rangle}\mathbf{d}\right) \\ &= \mathbf{T}\left[\ddot{\mathbf{r}} - \frac{2}{s\left\langle \mathbf{d}, \mathbf{v}\right\rangle}\left\langle \dot{\mathbf{r}}, \mathbf{v}\right\rangle \dot{\mathbf{r}}\right] = \mathbf{T}\left[\ddot{\mathbf{r}} - \alpha\dot{\mathbf{r}}\right]. \end{aligned} \tag{3.54}$$

■

In accordance with the decomposition of $\ddot{\mathbf{r}}$ in Eq. (3.37), we may also divide $\ddot{\mathbf{q}}$ into two terms

$$\ddot{\mathbf{q}} \;=\; \ddot{\mathbf{q}}_1 + \ddot{\mathbf{q}}_2, \tag{3.55}$$

where

$$\ddot{\mathbf{q}}_1 = \mathbf{T}\left[\ddot{\mathbf{r}}_1 - \frac{2}{s\left\langle \mathbf{d}, \mathbf{v}\right\rangle}\left\langle \dot{\mathbf{r}}, \mathbf{v}\right\rangle \dot{\mathbf{r}}\right], \quad \ddot{\mathbf{q}}_2 = \mathbf{T}\,\ddot{\mathbf{r}}_2. \tag{3.56}$$

As we can see, the third-order surface parameters e, f, g, h only appear in the second term $\ddot{\mathbf{q}}_2$.

3.3.2.3 Relationship Between $\dot{\mathbf{r}}$, $\ddot{\mathbf{r}}$ and κ^q

Proposition 7 The curvature κ^q of $\mathbf{q}(t)$ at $\mathbf{q}_0$ may be expressed as:

$$\kappa^q \;=\; \frac{s\left\langle \mathbf{d}, \mathbf{v}\right\rangle^3}{l}\frac{\left\langle \ddot{\mathbf{r}}, \mathbf{v}\right\rangle[\dot{\mathbf{r}},\, \mathbf{v}, \mathbf{d}] - \left\langle \dot{\mathbf{r}}, \mathbf{v}\right\rangle[\ddot{\mathbf{r}},\, \mathbf{v},\, \mathbf{d}] + \left\langle \mathbf{d}, \mathbf{v}\right\rangle[\ddot{\mathbf{r}},\, \mathbf{v},\, \dot{\mathbf{r}}]}{\left(||\,\dot{\mathbf{r}}\,||^2\left\langle \mathbf{d}, \mathbf{v}\right\rangle^2 + \left\langle \dot{\mathbf{r}}, \mathbf{v}\right\rangle^2 - 2\left\langle \dot{\mathbf{r}}, \mathbf{v}\right\rangle\left\langle \dot{\mathbf{r}}, \mathbf{d}\right\rangle\left\langle \mathbf{d}, \mathbf{v}\right\rangle\right)^{3/2}}. \tag{3.57}$$

Proof: For a planar curve $\mathbf{q}(t)$, its curvature and geodesic curvature are equivalent. Thus, from differential geometry [15] (Eq.(2.26), p.37)

$$\kappa^q \;=\; \frac{\langle \ddot{\mathbf{q}}, \mathbf{n}^q\rangle}{||\,\dot{\mathbf{q}}\,||^2}. \tag{3.58}$$

where Let $\mathbf{n}^q = \mathbf{v} \times \frac{\dot{\mathbf{q}}}{||\,\dot{\mathbf{q}}\,||}$. By taking the scalar product with $\mathbf{n}^q$ on both sides of Eq. (3.52), we get

$$\langle \ddot{\mathbf{q}}, \mathbf{n}^q \rangle \;=\; \gamma \langle \ddot{\mathbf{r}}, \mathbf{n}^q \rangle + 2\dot{\gamma} \langle \dot{\mathbf{r}}, \mathbf{n}^q \rangle + \ddot{\gamma} s \langle \mathbf{d}, \mathbf{n}^q \rangle . \tag{3.59}$$

By using Eq. (3.35), we may express the three dot products in Eq. (3.59) as follows:

$$\langle \ddot{\mathbf{r}}, \mathbf{n}^q \rangle \;=\; \frac{1}{||\,\dot{\mathbf{q}}\,||} \left(\gamma\,[\ddot{\mathbf{r}},\,\mathbf{v},\,\dot{\mathbf{r}}] + s\dot{\gamma}\,[\ddot{\mathbf{r}},\,\mathbf{v},\,\mathbf{d}]\right), \tag{3.60}$$

$$\langle \dot{\mathbf{r}}, \mathbf{n}^q \rangle \;=\; \frac{s\dot{\gamma}}{||\,\dot{\mathbf{q}}\,||}\,[\dot{\mathbf{r}},\,\mathbf{d},\,\mathbf{v}], \tag{3.61}$$

$$\langle \mathbf{d}, \mathbf{n}^q \rangle \;=\; \frac{\gamma}{||\,\dot{\mathbf{q}}\,||}\,[\mathbf{d},\,\mathbf{v},\,\dot{\mathbf{r}}]. \tag{3.62}$$

In the equations above, $[A,\,B,\,C]$ denotes the triple scalar product of three vectors. Substituting equations (3.60), (3.61) and (3.62) into Eq. (3.58) and then simplifying the result using equations (3.33), (3.34) and (3.53), we get

$$\begin{aligned}
\kappa^q \;&=\; \frac{1}{||\,\dot{\mathbf{q}}\,||^3}\left(s(\ddot{\gamma}\gamma - 2\dot{\gamma}^2)\,[\mathbf{d},\,\mathbf{v},\,\dot{\mathbf{r}}] + \gamma^2\,[\ddot{\mathbf{r}},\,\mathbf{v},\,\dot{\mathbf{r}}] + s\gamma\dot{\gamma}\,[\ddot{\mathbf{r}},\,\mathbf{v},\,\mathbf{d}]\right) \\
&=\; \frac{l^2}{s^2\,\langle \mathbf{d}, \mathbf{v}\rangle^3\,||\,\dot{\mathbf{q}}\,||^3}\left(\langle \ddot{\mathbf{r}}, \mathbf{v}\rangle\,[\dot{\mathbf{r}},\,\mathbf{v}, \mathbf{d}] - \langle \dot{\mathbf{r}}, \mathbf{v}\rangle\,[\ddot{\mathbf{r}},\,\mathbf{v},\,\mathbf{d}] + \langle \mathbf{d}, \mathbf{v}\rangle\,[\ddot{\mathbf{r}},\,\mathbf{v},\,\dot{\mathbf{r}}]\right),
\end{aligned} \tag{3.63}$$

where we have used the fact that $[\mathbf{d},\,\mathbf{v},\,\dot{\mathbf{r}}] = -\,[\dot{\mathbf{r}},\,\mathbf{v}, \mathbf{d}]$, and the identity

$$s(\ddot{\gamma}\gamma - 2\dot{\gamma}^2) \;=\; -\frac{l^2\,\langle \ddot{\mathbf{r}}, \mathbf{v}\rangle}{s^2\,\langle \mathbf{d}, \mathbf{v}\rangle^3}.$$

Furthermore, it follows from Eq. (3.35) that

$$\begin{aligned}
||\,\dot{\mathbf{q}}\,||^2 \;&=\; \langle \dot{\mathbf{q}}, \dot{\mathbf{q}}\rangle \;=\; \langle \gamma\dot{\mathbf{r}} + s\dot{\gamma}\mathbf{d}, \gamma\dot{\mathbf{r}} + s\dot{\gamma}\mathbf{d}\rangle \\
&=\; \gamma^2\,||\,\dot{\mathbf{r}}\,||^2 + s^2\dot{\gamma}^2 + 2s\gamma\dot{\gamma}\,\langle \mathbf{d}, \dot{\mathbf{r}}\rangle \\
&=\; \gamma^2\,||\,\dot{\mathbf{r}}\,||^2 + \gamma^2\frac{\langle \dot{\mathbf{r}}, \mathbf{v}\rangle^2}{\langle \mathbf{d}, \mathbf{v}\rangle^2} - 2\gamma^2\frac{\langle \dot{\mathbf{r}}, \mathbf{v}\rangle\,\langle \mathbf{d}, \dot{\mathbf{r}}\rangle}{\langle \mathbf{d}, \mathbf{v}\rangle} \\
&=\; \frac{\gamma^2}{\langle \mathbf{d}, \mathbf{v}\rangle^2}\left(||\,\dot{\mathbf{r}}\,||^2\,\langle \mathbf{d}, \mathbf{v}\rangle^2 + \langle \dot{\mathbf{r}}, \mathbf{v}\rangle^2 - 2\,\langle \dot{\mathbf{r}}, \mathbf{v}\rangle\,\langle \dot{\mathbf{r}}, \mathbf{d}\rangle\,\langle \mathbf{d}, \mathbf{v}\rangle\right).
\end{aligned}$$

Therefore, we obtain

$$||\,\dot{\mathbf{q}}\,|| \;=\; \frac{l}{s\,\langle \mathbf{d}, \mathbf{v}\rangle^2}\left(||\,\dot{\mathbf{r}}\,||^2\,\langle \mathbf{d}, \mathbf{v}\rangle^2 + \langle \dot{\mathbf{r}}, \mathbf{v}\rangle^2 - 2\,\langle \dot{\mathbf{r}}, \mathbf{v}\rangle\,\langle \dot{\mathbf{r}}, \mathbf{d}\rangle\,\langle \mathbf{d}, \mathbf{v}\rangle\right)^{1/2}. \tag{3.64}$$

By combining equations (3.64) and (3.63), we get the final expression (3.57) for κ^q. ■

Proposition 8 κ^q depends linearly on the third-order surface parameters.

Proof: $\dot{\mathbf{r}}$ in Eq. (3.9) and $\ddot{\mathbf{r}}_1$ in Eq. (3.37) are independent of the third-order surface parameters, and only the second term $\ddot{\mathbf{r}}_2$ in Eq. (3.37) of $\ddot{\mathbf{r}}$ depends linearly on e, f, g, h. Thus, the proposition follows from Eq. (3.57). ■

3.4 Properties of the Reflection Mapping

In this section we present several properties attached to first-order derivatives of the mapping $\mathbf{r}(t)$ when more than one line passes through the same reflection point. Such properties will be used in Section 3.5 when we study the inverse problem and we show how to recover surface position and shape from various types of image measurements. We start by introducing in Section 3.4.1 the concept of geometrical configuration as a geometrical structure defined by a pattern scene, an observer and the mirror surface. We say that the geometrical structure is *singular* if the mapping $\mathbf{r}(t)$ is not differentiable at the reflection point and *degenerate* if distinct coplanar scene lines are mapped into the same tangent direction after reflection. Then, we study in Section 3.4.2 and Section 3.4.3 the rank of the mapping when N distinct orientations of $\dot{\mathbf{r}}$ are considered (for both degenerate and non-degenerate configurations).

3.4.1 Geometrical Configurations

Let us assume a pattern scene composed of N coplanar lines $\mathbf{p}_1$, $\mathbf{p}_2$, $\cdots$ $\mathbf{p}_N$ through a point $\mathbf{p}_0$ [3]. As discussed in section 3.3, each scene line $\mathbf{p}_i$ is mapped into a reflected curve $\mathbf{r}_i(t)$ on the mirror surface and an image curve $\mathbf{q}_i(t)$ observed on the image plane, with their tangent vectors at t_0 denoted by $\dot{\mathbf{r}}_i$ and $\dot{\mathbf{q}}_i$ respectively. Let ϕ_i be the angle between $\dot{\mathbf{r}}_i$ and the $\mathbf{u}$ axis at $\mathbf{r}_0$, and let ψ_i be the angle between $\dot{\mathbf{q}}_i$ and horizontal axis in the image plane at $\mathbf{q}_0$. It then follows from Eq. (3.26) that the tangent direction $\tan\phi_i$ of $\mathbf{r}_i(t)$ can be computed by

$$\tan\phi_i = \frac{(J_u - 2a\cos\theta)B_{v_i} + 2cB_{u_i}\cos\theta}{(J_v - 2b\cos\theta)B_{u_i} + 2cB_{v_i}\cos\theta}. \tag{3.65}$$

[3] The hypothesis of coplanar scene lines is necessary for the ensuing development. See in particular Proposition 13.

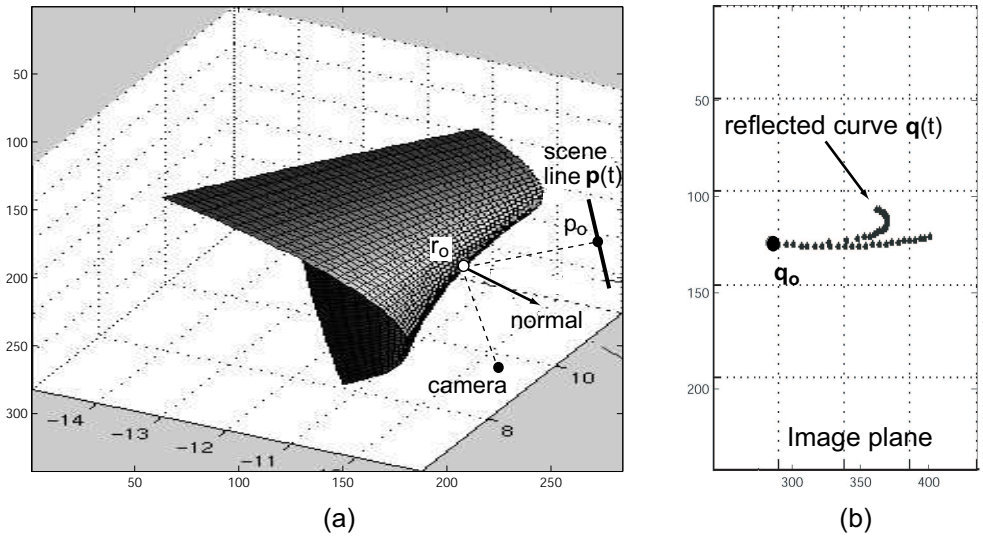


Figure 3.3: **Example of singular geometrical configuration**: (a) A line $\mathbf{p(t)}$ passing through $\mathbf{p}_0$ is reflected off a paraboloid mirror and observed by a camera. The curvature of the paraboloid at $\mathbf{r}_0$ is such that the corresponding geometrical configuration is singular. (b) The reflected image curve is shown. Notice the discontinuity of mapping $\mathbf{q(t)}$ at $\mathbf{q}_0$.

The corresponding tangent direction $\tan\psi_i$ of $\mathbf{q}_i(t)$ in the image plane can be related to $\tan\phi_i$ by means of Proposition 4.

Definition 1 We call a *geometrical configuration* $\wp = \wp(\ \mathbf{c}, \mathbf{p}_0, \mathbf{p}_1, \mathbf{p}_2, \cdots\ \mathbf{p}_n\ \mathbf{r}_0, \mathbf{n}_r, a, b, c)$ a geometrical structure where $\mathbf{p}_0$ is a a scene point, $\mathbf{c}$ is an observer, $\mathbf{r}_0$ is the corresponding reflected point on a mirror surface S, $\mathbf{n}_r$ is the normal of S at $\mathbf{r}_0$, the quantities a, b, c are the second-order parameters of S around $\mathbf{r}_0$ in the principal plane reference system and $\mathbf{p}_1$, $\mathbf{p}_2$, $\cdots\ \mathbf{p}_n$ are n coplanar distinct scene lines passing through $\mathbf{p}_0$.

3.4.1.1 Singular Configurations

Definition 2 We call a geometrical configuration *singular* if the condition (3.19) for the existence of the mapping $\mathbf{r}(t)$ around the neighborhood of $\mathbf{r}_0$ is not satisfied (i.e., if the Jacobian $\Delta = 0$).

Notice that if $\Delta = 0$ the mapping $\mathbf{r}_i(t)$ is not differentiable at t_0. An example of singular geometrical configuration is shown in Figure 3.3.

Proposition 9 A geometrical configuration $\wp$ is singular if and only if the condition

$$c^2 = (\frac{J_u}{2\cos\theta} - a)(\frac{J_v}{2\cos\theta} - b) \tag{3.66}$$

holds.

Proof: from Eq. (3.3.1.1), condition (3.66) holds if and only if $\Delta = 0$. ■

Proposition 10 If the surface is a convex paraboloid in the neighborhood of the reflection point, then the corresponding geometrical configuration is non-singular.

Proof: If the surface is a convex paraboloid in the neighborhood of the reflection point, the second-order parameters a, b and c satisfy:

$$\begin{cases} a < 0; \quad b < 0 \\ c^2 < ab \end{cases} \tag{3.67}$$

Let us assume by contradiction that the geometrical configuration is singular. Since $J_u, J_v > 0$ and the reflection angle $\theta < \pi/2$,

$$c^2 < ab < (\frac{J_u}{2\cos\theta} - a)(\frac{J_v}{2\cos\theta} - b) \tag{3.68}$$

which contradicts Eq. (3.66) ■

3.4.1.2 Degenerate Configurations

Definition 3 We call a geometrical configuration *degenerate* if for any $i, j(i \neq j)$, we have $\phi_i = \phi_j$.

Proposition 11 A geometrical configuration $\wp$ is degenerate if and only if all the scene lines $\mathbf{p}_1, \mathbf{p}_2, \cdots \mathbf{p}_n$ define a plane passing through $\mathbf{p}_0$ and $\mathbf{r}_0$.

Proof: We divide the proof in two steps: 1) we prove that for any $i, j(i \neq j)$, $\phi_i = \phi_j$ if and only if $\frac{B_{u_i}}{B_{v_i}} = \frac{B_{u_j}}{B_{v_j}}$; 2) we prove that $\frac{B_{u_i}}{B_{v_i}} = \frac{B_{u_j}}{B_{v_j}} = k$ if and only if $\mathbf{p}_i$, $\mathbf{p}_j$ define a plane passing through $\mathbf{p}_0$ and $\mathbf{r}_0$.

1. (←) Suppose $B_{u_i}/B_{v_i} = B_{u_j}/B_{v_j} = k$, it follows from Eq. (3.65) that

$$\tan\phi_i = \tan\phi_j = \frac{(J_u - 2a\cos\theta)k + 2c\cos\theta}{(J_v - 2b\cos\theta) + 2kc\cos\theta}. \tag{3.69}$$

(→) From Eq. (3.65):

$$\frac{B_{u_i}}{B_{v_i}} = \frac{(J_u - 2a\cos\theta) + 2c\tan\phi_i\cos\theta}{(J_v - 2b\cos\theta)\tan\phi_i + 2c\cos\theta}. \tag{3.70}$$

Then $B_{u_i}/B_{v_i} = B_{u_j}/B_{v_j}$ follows immediately from $\phi_i = \phi_j$.

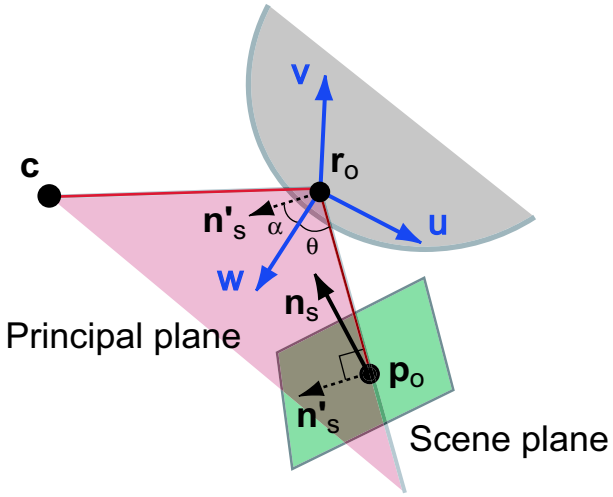


Figure 3.4: **Sketch of the proof of Proposition 11 (step 2)**: Condition $(\mathbf{n_s})_w = -\tan\theta(\mathbf{n_s})_u$ is equivalent to have the scene lines contained in a plane which also contains $\mathbf{p}_0 - \mathbf{r}_0$. $\mathbf{n_s}$ is the normal vector of the scene plane. $\mathbf{n_s}'$ is the projection of $\mathbf{n_s}$ into the principal plane. $\mathbf{n_s}'$ is orthogonal to $\mathbf{p}_0 - \mathbf{r}_0$ if and only if $\alpha = 90 - \theta$, and $(\mathbf{n_s})_w = -\tan\theta(\mathbf{n_s})_u$

2. It follows from Eq. (3.10) that if $k = \infty$ (or equivalently, $B_{v_i} = B_{v_j} = 0$), $\mathbf{p}_i$ and $\mathbf{p}_j$ belong to the principal plane which passes through $\mathbf{p}_0$ and $\mathbf{r}_0$. If $k \neq \infty$, we have

$$\frac{B_{u_i}}{B_{v_i}} = \frac{B_{u_j}}{B_{v_j}} = k \quad \Leftrightarrow$$

$$\frac{(\Delta\mathbf{p}_i)_w \sin\theta - (\Delta\mathbf{p}_i)_u \cos\theta}{(\Delta\mathbf{p}_i)_v} = \frac{(\Delta\mathbf{p}_j)_w \sin\theta - (\Delta\mathbf{p}_j)_u \cos\theta}{(\Delta\mathbf{p}_j)_v} = \frac{-k}{\cos\theta} \quad \Leftrightarrow$$

$$\tan\theta = \frac{(\Delta\mathbf{p}_i)_u(\Delta\mathbf{p}_j)_v - (\Delta\mathbf{p}_j)_u(\Delta\mathbf{p}_i)_v}{(\Delta\mathbf{p}_i)_w(\Delta\mathbf{p}_j)_v - (\Delta\mathbf{p}_j)_w(\Delta\mathbf{p}_i)_v} = -\frac{(\mathbf{n_s})_w}{(\mathbf{n_s})_u}$$

where $\mathbf{n_s} = [(\mathbf{n_s})_u \ (\mathbf{n_s})_v \ (\mathbf{n_s})_w]^T$ is the normal vector of the plane formed by the scene lines $\mathbf{p}_i$ and $\mathbf{p}_j$. $(\mathbf{n_s})_w = -\tan\theta(\mathbf{n_s})_u$ suggests that $\mathbf{n_s}$ must be orthogonal to $\mathbf{p}_0 - \mathbf{r}_0$ (see Figure 3.4). By construction, the plane defined by $\mathbf{p}_i$ and $\mathbf{p}_j$ passes through $\mathbf{p}_0$.

■

Notice that in the special case where $\mathbf{p}_i$, $\mathbf{p}_j$ belongs to the principal plane (hence, $k = \infty$), we have from Eq. (3.65) that

$$\tan\phi_i = \tan\phi_j = \frac{2c\cos\theta}{(J_v - 2b\cos\theta)}. \tag{3.71}$$

As we shall see in Section 3.5, if a geometrical configuration is degenerate, the reconstruction of the surface around $\mathbf{r}_0$ is not feasible. In particular, having two or more scene lines contained in the principal plane leads to inaccurate reconstruction results (see Section 3.5.3.3 or Figure 3.15 for details).

3.4.2 The Rank Theorem

In this section we study the rank of the mapping when N distinct orientations of $\dot{\mathbf{r}}$ are considered (for both degenerate and non-degenerate configurations). We show that if the configuration is degenerate, the rank of the mapping is 1, whereas if the configuration is non-degenerate, the rank of the mapping is 2. We group these properties under the name of *Rank Theorem*.

By considering n $(n \geq 3)$ distinct coplanar scene lines intersecting at $\mathbf{p}_0$, we may obtain the following linear system about the 3 parameters a, b, c. In fact, let

$$\begin{aligned} \alpha &= J_u - 2a\cos\theta, \\ \beta &= J_v - 2b\cos\theta, \\ \gamma &= 2c\cos\theta, \end{aligned}$$

then Eq. (3.65) can be represented as

$$\begin{bmatrix} B_{v_i} & -B_{u_i}\tan\phi_i & B_{u_i} - B_{v_i}\tan\phi_i \end{bmatrix} \begin{bmatrix} \alpha \\ \beta \\ \gamma \end{bmatrix} = 0,$$

denoted by

$$\mathbf{H}_n\,\mathbf{g} = 0, \tag{3.72}$$

where

$$\mathbf{g} = [J_u - 2a\cos\theta \;\; J_v - 2b\cos\theta \;\; 2c\cos\theta],^{\mathrm{T}} \tag{3.73}$$

and $\mathbf{H}_n$ is a $n \times 3$ matrix defined as

$$\mathbf{H}_n = \begin{bmatrix} B_{v_1} & -B_{u_1}\tan\phi_1 & B_{u_1} - B_{v_1}\tan\phi_1 \\ B_{v_2} & -B_{u_2}\tan\phi_2 & B_{u_2} - B_{v_2}\tan\phi_2 \\ \vdots & \vdots & \vdots \\ B_{v_n} & -B_{u_n}\tan\phi_n & B_{u_n} - B_{v_n}\tan\phi_n \end{bmatrix}, \tag{3.74}$$

In the following, we show some properties of $Rank(\mathbf{H}_n)$.

Proposition 12 Given a non-singular geometrical configuration $\wp$, the $Rank(\mathbf{H}_n)$ is always different from zero.

Proof: Suppose by contradiction that $Rank(\mathbf{H}_n) = 0$. Then, each entry of $\mathbf{H}_n$ is zero. Hence, $B_{v_i} = B_{u_i} = 0$ for $\forall i$. Thus, all scene lines lie in the principal plane formed by $\mathbf{c}, \mathbf{r}_0, \mathbf{p}_0$ and satisfy the constraint $(\Delta\mathbf{p}_i)_w \cos\theta \sin\theta = (\Delta\mathbf{p}_i)_u \cos^2\theta$, namely $(\Delta\mathbf{p}_i)_u/(\Delta\mathbf{p}_i)_w = \tan\theta$. As a result, each scene line is oriented along the incident vector $\mathbf{r}_0 - \mathbf{p}_0$, contradicting the underlying assumption of n distinct scene lines. ■

Proposition 13 Given a non-singular geometrical configuration $\wp$, $Rank(\mathbf{H}_n) = 1$ for $n \geq 3$, if and only if $\wp$ is degenerate.

Proof: We denote the row vectors of $\mathbf{H}_n$ by $\mathbf{h}_1, \mathbf{h}_2, \ldots, \mathbf{h}_n$.

1. ($\rightarrow$) If $Rank(\mathbf{H}_n) = 1$, any 2 rows $\mathbf{h}_i$ and $\mathbf{h}_j$ of $\mathbf{H}_n$ are linearly dependent, namely there exists a scalar $\alpha \neq 0$ such that $\mathbf{h}_j = \alpha\mathbf{h}_i$. Thus,

$$\begin{aligned} B_{v_j} &= \alpha\, B_{v_i}, & (1) \\ B_{u_j}\tan\phi_j &= \alpha\, B_{u_i}\tan\phi_i, & (2) \\ B_{u_j} - B_{v_j}\tan\phi_j &= \alpha B_{u_i} - \alpha B_{v_i}\tan\phi_i, & (3) \end{aligned}$$

   Substituting (1) and (2) into (3), yields the following equations:

$$\begin{aligned} (\tan\phi_i - \tan\phi_j)(B_{u_i} - B_{v_i}\tan\phi_j) &= 0 \\ (\tan\phi_i - \tan\phi_j)(B_{u_j} + B_{v_j}\tan\phi_i) &= 0 \end{aligned}$$

   which is satisfied when either $\phi_i = \phi_j$ or $\tan\phi_i = -B_{u_j}/B_{v_j}$ and $\tan\phi_j = B_{u_i}/B_{v_i}$. If $\phi_i = \phi_j$, it follows from Proposition 11 (see proof, step 2) that $\mathbf{p}_i$ and $\mathbf{p}_j$ defines a plane passing through $\mathbf{p}_0$

and $\mathbf{r}_0$. Since the scene lines are coplanar from Definition 1, it follows immediately that all the scene lines fall in this same plane passing through $\mathbf{r}_0$, which leads to a degenerate configuration according to Proposition 11. In fact, if there exists a pair of rows $\mathbf{h}_i$ and $\mathbf{h}_j$ such that $\phi_i = \phi_j$, $\wp$ is degenerate. If $\tan\phi_i = -B_{u_j}/B_{v_j}$ and $\tan\phi_j = B_{u_i}/B_{v_i}$, by considering another pair of rows $\mathbf{h}_i$ and $\mathbf{h}_k$ and following the similar derivations, we can get another pair of constraints, that is, $\tan\phi_i = -B_{u_k}/B_{v_k}$ and $\tan\phi_k = B_{u_i}/B_{v_i}$. Therefore, we have $\phi_j = \phi_k$, again leading to a degenerate configuration due to the reason mentioned above.

2. ($\leftarrow$) If the geometrical configuration is degenerate, then according to Definition 3 and Proposition 11 (see proof, step 1) , $\exists k$, such that $\forall i, j$, $B_{u_i}/B_{v_i} = B_{u_j}/B_{v_j} = k$. It follows that $\mathbf{h}_j = \mathbf{h}_i * k * B_{u_j}/B_{v_i}$. Thus, $Rank(\mathbf{H}_n) = 1$, since $Rank(\mathbf{H}_n) \neq 0$ is guaranteed by Proposition 12. ■

Proposition 14 Given a non-singular geometrical configuration $\wp$, $Rank(\mathbf{H}_n) = 2$ for $n \geq 3$, if and only if $\wp$ is non-degenerate.

Proof: We denote the row vectors of $\mathbf{H}_n$ by $\mathbf{h}_1, \mathbf{h}_2, \ldots, \mathbf{h}_n$.

1. ($\leftarrow$) Since the row rank and the column rank of a matrix are equal, $Rank(\mathbf{H}_n)$ can only have values of $0, 1, 2, 3$. It follows immediately from Proposition 12 and Proposition 13 that $Rank(\mathbf{H}_n) = 2$ or $Rank(\mathbf{H}_n) = 3$. If $Rank(\mathbf{H}_n) = 3$, then there exists three independent row vectors among $\mathbf{h}_1, \mathbf{h}_2, \ldots, \mathbf{h}_n$. Without loss of generality, we assumes that they are $\mathbf{h}_1, \mathbf{h}_2, \mathbf{h}_3$. It then follows from Eq. (3.72) that

$$\begin{bmatrix} B_{v_1} & -B_{u_1}\tan\phi_1 & B_{u_1} - B_{v_1}\tan\phi_1 \\ B_{v_2} & -B_{u_2}\tan\phi_2 & B_{u_2} - B_{v_2}\tan\phi_2 \\ B_{v_3} & -B_{u_3}\tan\phi_3 & B_{u_3} - B_{v_3}\tan\phi_3 \end{bmatrix} \begin{bmatrix} \alpha \\ \beta \\ \gamma \end{bmatrix} = \mathbf{0},$$

which leads to a zero solution, that is, $\alpha = \beta = \gamma = 0$. Consequently, according to Proposition 9, $\wp$ is singular, contradictory to the assumption.

2. ($\rightarrow$) Proof by contradiction. If $\wp$ is degenerate, we will have $Rank(\mathbf{H}_n) = 1$ by means of Proposition 13. ■

We combine Proposition 12, Proposition 13 and Proposition 14 under the name of *Rank Theorem*.

3.4.3 The Generalized Rank Theorem for Arbitrary Tangent Directions

In this section we study the rank of the matrix $\mathbf{H}_n$ when N distinct arbitrary values of $\phi_1, \phi_2, \ldots, \phi_n$ are considered. We show that if the configuration is degenerate, the rank of $\mathbf{H}_n$ is 2, whereas if the configuration is non-degenerate, the rank of $\mathbf{H}_n$ is 3. We group these properties under the name of *Generalized Rank Theorem.*

As discussed in Section 3.2, we refer to parameter s (namely, the distance between $\mathbf{c}$ and actual reflecting point $\mathbf{r}_0$) as the first-order surface parameter which determines the position $\mathbf{r}_0$ and normal $\mathbf{n}_r$ of the surface (by means of Eq. (3.1) and Eq. (3.5)). Thus, as the first-order parameter s varies, a family of geometrical configurations $\wp(s)$ can be defined (here, $\mathbf{c}$, $\mathbf{p}_0$ $\mathbf{p}_1$, $\mathbf{p}_2$, $\cdots$ $\mathbf{p}_n$, a, b, c are fixed).

Let $\mathbf{\Psi^m} = [\psi_1^m, \psi_2^m, \ldots \psi_n^m]$ denote the set of tangent directions measured at $\mathbf{q}_0$ for the image curves $\mathbf{q}_1, \mathbf{q}_2, \ldots, \mathbf{q}_n$ (images of the n scene lines $\mathbf{p}_1, \mathbf{p}_2, \ldots \mathbf{p}_n$ intersecting at $\mathbf{p}_0$) for a geometrical configuration $\wp(s)$. Let $\hat{\mathbf{\Psi}}$ denote an arbitrary set of tangent directions $[\hat{\psi}_1, \hat{\psi}_2, \ldots, \hat{\psi}_n]$ at $\mathbf{q}_0$.

Definition 4 We say that the set of directions $\hat{\mathbf{\Psi}}$ is *compatible* with a geometrical configuration at s if $\wp(s)$ yields a set of measurements $\mathbf{\Psi^m}$ equal to $\hat{\mathbf{\Psi}}$. Otherwise, $\hat{\mathbf{\Psi}}$ is *not compatible* with $\wp(s)$.

Let $[\hat{\phi}_1, \hat{\phi}_2, \ldots, \hat{\phi}_n]$ denote the projection of $\hat{\mathbf{\Psi}}$ into the surface tangent plane at $\mathbf{r}_0$. Thus, we may define a matrix $\hat{\mathbf{H}}_n$ with a similar form as $\mathbf{H}_n$ in Eq. (3.74). That is,

$$\hat{\mathbf{H}}_n = \begin{bmatrix} B_{v_1} & -B_{u_1}\tan\hat{\phi}_1 & B_{u_1} - B_{v_1}\tan\hat{\phi}_1 \\ B_{v_2} & -B_{u_2}\tan\hat{\phi}_2 & B_{u_2} - B_{v_2}\tan\hat{\phi}_2 \\ \vdots & \vdots & \vdots \\ B_{v_n} & -B_{u_n}\tan\hat{\phi}_n & B_{u_n} - B_{v_n}\tan\hat{\phi}_n \end{bmatrix}. \tag{3.75}$$

Obviously, if $\hat{\mathbf{\Psi}} = \mathbf{\Psi^m}$, then $\hat{\mathbf{H}}_n = \mathbf{H}_n$. In the following, we shall study the rank property of $\hat{\mathbf{H}}_n$, and the result will be used in Section 3.5.

Proposition 15 Consider a geometrical configuration $\wp(s)$, an arbitrary set of tangent directions $\hat{\mathbf{\Psi}} = [\hat{\psi}_1, \hat{\psi}_2, \ldots, \hat{\psi}_n]$ and its corresponding projections $[\hat{\phi}_1, \hat{\phi}_2, \ldots, \hat{\phi}_n]$ on the tangent plane. Assume that tangent orientations are distinct (i.e., $\forall i, j (i \neq j)$, $\hat{\phi}_i \neq \hat{\phi}_j$). Then, $Rank(\hat{\mathbf{H}}_n) \geq 2$.

Proof: We want to prove that $Rank(\hat{\mathbf{H}}_n)$ can be neither 0 nor 1 by contradiction.

1. If $Rank(\hat{\mathbf{H}}_n) = 0$, all scene lines $\mathbf{p}_1$, ..., $\mathbf{p}_N$ are aligned, which contradicts Definition 1.
2. If $Rank(\hat{\mathbf{H}}_n) = 1$, using a proof similar to that of Proposition 13, one can show that the hypothesis of distinct tangent orientations is violated. ■

Proposition 16 Consider a geometrical configuration $\wp(s)$, an arbitrary set of tangent directions $\hat{\mathbf{\Psi}} = [\hat{\psi}_1, \hat{\psi}_2, \ldots, \hat{\psi}_n]$ and its corresponding projections $[\hat{\phi}_1, \hat{\phi}_2, \ldots, \hat{\phi}_n]$ on the tangent plane. Assume that tangent orientations are distinct (i.e., $\forall i, j (i \neq j)$, $\hat{\phi}_i \neq \hat{\phi}_j$). Additionally, assume that $\hat{\mathbf{\Psi}}$ is *not* compatible with $\wp(s)$. Then, $Rank(\hat{\mathbf{H}}_n) = 2$ if and only if $\wp(s)$ is degenerate.

Proof:

1. ($\leftarrow$) We want to show that if $\wp(s)$ is degenerate, then $Rank(\hat{\mathbf{H}}_n) = 2$. It follows from Definition 3 and Proposition 11 (see proof, step 1) that $\exists k$, $B_{u_i} = kB_{v_i}$ $\forall i \neq j$. If $k \neq \infty$, we may express $\hat{\mathbf{H}}_n$ as follows:

$$\hat{\mathbf{H}}_n = \hat{\mathbf{H}}_n^1 \hat{\mathbf{H}}_n^2 = \begin{bmatrix} B_{u_1} & 0 & \ldots & 0 \\ 0 & B_{u_2} & \ldots & 0 \\ & \vdots & & \\ 0 & 0 & \ldots & B_{u_n} \end{bmatrix} \begin{bmatrix} \frac{1}{k} & -\tan\hat{\phi}_1 & 1 - \frac{1}{k}\tan\hat{\phi}_1 \\ \frac{1}{k} & -\tan\hat{\phi}_2 & 1 - \frac{1}{k}\tan\hat{\phi}_2 \\ & \vdots & \\ \frac{1}{k} & -\tan\hat{\phi}_n & 1 - \frac{1}{k}\tan\hat{\phi}_n \end{bmatrix}$$

   Hence, $Rank(\hat{\mathbf{H}}_n) = Rank(\hat{\mathbf{H}}_n^1 \hat{\mathbf{H}}_n^2) \leq \min(Rank(\hat{\mathbf{H}}_n^1), Rank(\hat{\mathbf{H}}_n^2))$. Since the third column of $\hat{\mathbf{H}}_n^2$ can be expressed as a linear combination of the first two, $Rank(\hat{\mathbf{H}}_n^2) \leq 2$, and thus, $Rank(\hat{\mathbf{H}}_n) \leq 2$. On the other hands, we have $Rank(\hat{\mathbf{H}}_n) \geq 2$ from Proposition 15. Therefore, $Rank(\hat{\mathbf{H}}_n) = 2$. Furthermore, if $k = \infty$,

$$\hat{\mathbf{H}}_n = \hat{\mathbf{H}}_n^1 \hat{\mathbf{H}}_n^2 = \begin{bmatrix} B_{u_1} & 0 & B_{u_1}\tan\hat{\phi}_1 \\ B_{u_2} & 0 & B_{u_2}\tan\hat{\phi}_2 \\ & \vdots & \\ B_{u_n} & 0 & B_{u_n}\tan\hat{\phi}_n \end{bmatrix}$$

   which leads again to $Rank(\hat{\mathbf{H}}_n) = 2$.

2. ($\rightarrow$) We want to show that if $Rank(\hat{\mathbf{H}}_n) = 2$, then $\wp(s)$ is degenerate. Assume by contradiction that $\wp(s)$ is non-degenerate. Consider the system $\hat{\mathbf{H}}_n \mathbf{g} = 0$ in Eq. (3.72). Since $Rank(\hat{\mathbf{H}}_n) = 2$, vector $\mathbf{g}$ is perpendicular to the plane spanned by any two rows of $\hat{\mathbf{H}}_n$. That is, $\mathbf{g} = r\ \mathbf{h}$, where r is a scalar parameter and $\mathbf{h} = [h_1 \ \ h_2 \ \ h_3]^{\mathrm{T}} = \mathbf{h}_k \times \mathbf{h}_j$, where $\mathbf{h}_k, \mathbf{h}_j$ are the kth and jth rows of $\mathbf{H}$ associated with 2 arbitrary lines $\mathbf{p}_k$ and $\mathbf{p}_j$. Since $\mathbf{g} = [J_u - 2a\cos\theta \ J_v - 2b\cos\theta \ 2c\cos\theta]^{\mathrm{T}}$, we have found a family of values for a, b, c and a corresponding family of non-degenerate geometrical configurations

$\wp(s, a, b, c)$ which are parameterized by r and yield the set of tangent orientations $\hat{\mathbf{\Psi}}$, contradicting the non-compatibility assumption iii).

■

Proposition 17 Consider a geometrical configuration $\wp(s)$, an arbitrary set of tangent directions $\hat{\mathbf{\Psi}} = [\hat{\psi}_1, \hat{\psi}_2, \ldots, \hat{\psi}_n]$ and its corresponding projections $[\hat{\phi}_1, \hat{\phi}_2, \ldots, \hat{\phi}_n]$ on the image plane. Assume that tangent orientations are distinct (i.e., $\forall i, j (i \neq j)$, $\hat{\phi}_i \neq \hat{\phi}_j$). Additionally, assume that $\hat{\mathbf{\Psi}}$ is *not* compatible with $\wp(s)$. Then, $Rank(\hat{\mathbf{H}}_n) = 3$ if and only if $\wp(s)$ is non-degenerate.

Proof:

1. ($\leftarrow$) If $\wp(s)$ is non-degenerate, but $Rank(\hat{\mathbf{H}}_n) \neq 3$. By Proposition 15, $Rank(\hat{\mathbf{H}}_n) \geq 2$ and since $Rank(\hat{\mathbf{H}}_n) \leq 3$, $Rank(\hat{\mathbf{H}}_n) = 2$. It then follows from Proposition 16 that $\wp(s)$ must be degenerate, contradicting our assumption.

2. ($\rightarrow$) Proof by contradiction. If $\wp$ is degenerate, then it follows from Proposition 16 that $Rank(\hat{\mathbf{H}}_n) = 2$, contradicting the assumption $Rank(\hat{\mathbf{H}}_n) = 3$.

■

We combine Proposition 15, Proposition 16 and Proposition 17 under the name of *Generalized Rank Theorem*.

To sum it all up, we have listed the results obtained in this section in Table 3.3.

$Rank(\hat{\mathbf{H}}_n)$	$\wp(s)$ deg.	$\wp(s)$ non-deg.	Pointer
$\mathbf{\Psi}$ compatible with $\wp(s)$	1	2	Proposition 13 Proposition 14
$\mathbf{\Psi}$ not compatible with $\wp(s)$	2	3	Proposition 16 Proposition 17

Table 3.3: Rank property of $\hat{\mathbf{H}}_n$ for arbitrary tangent direction.

3.5 The Inverse Problem

3.5.1 Inverse Problem

In this section, we study the inverse problem and apply the analytical formulas derived in Section 3.3 to reconstruct the geometry of a mirror surface by measuring its deforming effects on a scene planar grid.

We start by showing in Section 3.5.2 that if the measurements of orientation and position of (at least) two reflected curves are available, the parameters describing local shape of the surface can be reduced. Then, we demonstrate in Section 3.5.3 and Section 3.5.4 that by exploiting different types of image measurements and by assuming that the scene is calibrated, we may recover the shape of the mirror surface to different accuracy. We take advantage of the results proven in Section 3.4 to study degenerate solutions and ambiguities of the reconstruction problem.

We present two reconstruction algorithms. The first algorithm (call it A1) assumes that the orientation of at least 3 lines are available at the same intersection point and estimates surface position and orientation at the reflection point. Surface curvature and the third-order surface parameters can only be recovered up to a free parameter. Surface curvature can be fully estimated only when the third surface parameters are negligible (see Figure 3.5).

The second algorithm (A2) assumes that orientation and scale of at least 2 lines are available at the same intersection point. Scale information measures the "velocity" along the line in the image, assuming known velocity along the line in the scene. The algorithm estimates surface position and surface orientation at the reflection point. Additionally, it estimates surface curvature and the third-order surface parameters in closed-form (see Figure 3.10). These results are summarized in Table 3.4.

Algorithm	**Measurements**	**Estimated surface quantities**
A1	point $\mathbf{q}_0$ + orientation of 3 lines through $\mathbf{q}_0$	distance, tangent plane at $\mathbf{r}_0$
A2	point $\mathbf{q}_0$ + orientation & scale of 2 lines through $\mathbf{q}_0$	distance, tangent plane, curvature and 3^{rd}-order parameters at $\mathbf{r}_0$

Table 3.4: Comparison of the two algorithms

Finally, in Section 3.5.4.5 we show that the hypothesis on the structure of the scene (i.e., scene composed of a grid of intersecting lines) is not necessary. We demonstrate that scale and orientation measurements may also be extracted from the reflection of a planar scene patch of arbitrary geometry. We prove that local surface shape can be estimated if the location of at least 3 arbitrary points is available within a neighborhood of the reflected scene patch.

For clarity, throughout this section, we indicate the quantities measured in the image plane with the superscript m and the quantities associated with different scene lines with a subscript. For example, the measurement of the first-order derivative of the ith curve $\mathbf{q}_i(t)$ in the image plane is indicated by $\dot{\mathbf{q}}_i^m$.

3.5.2 Parameter Reduction

In this section, we show that if the measurements of the orientation of (at least) two reflected curves and the position of the point of intersection are available, the parameters describing the surface position, orientation and curvature (i.e., s, a, b, c) can be reduced from 4 to 2.

As discussed in Section 3.2, we refer to parameter s as the distance between $\mathbf{c}$ and actual reflecting point $\mathbf{r}_0$. Denote by $\mathbf{\Psi^m} = [\psi_1^m, \psi_2^m, \ldots \psi_n^m]$ the set of tangent directions measured at $\mathbf{q}_0$ for the image curves $\mathbf{q}_1$, $\mathbf{q}_2, \ldots, \mathbf{q}_n$ (images of the n scene lines $\mathbf{p}_1, \mathbf{p}_2, \ldots \mathbf{p}_n$ intersecting at $\mathbf{p}_0$). We may construct the $n \times 3$ matrix $\mathbf{H}$ and the vector $\mathbf{g}$ in Eq. (3.72) from measurements ($\mathbf{\Psi^m}$ and $\mathbf{q}_o^m$) and known geometrical quantities from calibration. Notice that in Eq. (3.72), measurements and second surfaces parameters are separated. In fact, tangent measurements are contained in $\mathbf{H}$ only and second-order surface parameters (a, b, c) are contained in $\mathbf{g}$ only. Both $\mathbf{H}$ and $\mathbf{g}$ are a function of s.

Proposition 14 suggests that given a non-singular geometrical configuration, the vector $\mathbf{g}$ in Eq. (3.73) is perpendicular to the plane spanned by any two rows of $\mathbf{H}$. That is, $\mathbf{g} = r\ \mathbf{h}$, where r is an unknown scalar parameter and $\mathbf{h} = [h_1 \quad h_2 \quad h_3]^{\mathrm{T}} = \mathbf{h}_k \times \mathbf{h}_j$. Here, $\mathbf{h}_k, \mathbf{h}_j$ are the kth and jth rows of $\mathbf{H}$ associated with 2 arbitrary lines $\mathbf{p}_k$ and $\mathbf{p}_j$. As a result, only two lines suffice to univocally constrain a, b, c as a function of r and s. That is,

$$\left\{ \begin{array}{rcl} a & = & \dfrac{J_u}{2\cos\theta} - r\dfrac{h_1}{2\cos\theta} \\ b & = & \dfrac{J_v}{2\cos\theta} - r\dfrac{h_2}{2\cos\theta} \\ c & = & r\dfrac{h_3}{2\cos\theta} \end{array} \right. , \tag{3.76}$$

which reduces the total number of unknowns from 8 (s, a, b, c, e, f, g, h) to 6 (s, r, e, f, g, h). Note that r appears as a free parameter.

3.5.3 Image Measurements: Curve Orientations

In Section 3.5.3.1, we show that by measuring position and tangent orientations of the reflection images of (at least) three scene lines intersecting at a point $\mathbf{p}_0$, we are able to recover the geometry of a mirror surface up to first-order (position and normal). In Section 3.5.3.2, we give an explicit reconstruction algorithm which may be summarized in 8 steps. In Section 3.5.3.4, we discuss the possibility of having multiple or degenerate solutions. In Section 3.5.3.4, we show that the second-order surface parameters can only be recovered up to one free parameter r and we refer to it as *second-order parameter ambiguity*. In Section 3.5.3.5, we present explicit solutions for the second-order surface parameters for two special surfaces: the sphere and

the cylinder. In Section 3.5.3.6, we attack the general case. We show that if the curvature of (at least) one reflected curve at $\mathbf{q}_0$ is available, the second-order parameter ambiguity can be resolved and r can be derived in closed form solution. This derivation requires the assumption that third-order surface parameters e, f, g, h are negligible in the neighborhood of $\mathbf{r}_0$. Additionally, we prove (see Proposition 19) that the image curvature at $\mathbf{q}_0$ depends linearly asymptotically on r and that the slope of such a function does not depend on the third-order surface parameters. Proposition 19 may be used to estimate the second-order parameters in closed form. Finally, we apply these results in Section 3.5.3.7 where we give expressions to constrain the third-order surface parameter e, f, g, h. We show that if the measurements of (at least) four reflected image curvatures at $\mathbf{q}_0$ are available, then e, f, g, h can be estimated up to r. We refer to it as *third-order parameter ambiguity*.

3.5.3.1 Recovering the First-Order Parameters

By constructing the $n \times 3$ matrix $\mathbf{H}$ in Eq. (3.74) from measurements ($\mathbf{\Psi^m}$ and $\mathbf{q}_o^m$) and known geometrical quantities, we obtain $\det(\mathbf{H}^\mathrm{T}\mathbf{H})$ as a parametric function of s. Consider n scene lines ($n \geq 3$) and denote by s^* the actual distance between $\mathbf{c}$ and unknown reflecting point $\mathbf{r}_0$. Since $Rank(\mathbf{H}) = Rank(\mathbf{H}^\mathrm{T}\mathbf{H})$, Proposition 14 suggests a necessary condition for s^*. Namely, $\det(\mathbf{H}^\mathrm{T}\mathbf{H})$ must vanish at s^*. If $\det(\mathbf{H}^\mathrm{T}\mathbf{H}) = 0$ has a single root, then we find s^* univocally. However, there might exist some values of $s \neq s^*$ such that $\det(\mathbf{H}^\mathrm{T}\mathbf{H}) = 0$. Considering a geometrical configuration $\wp(s)$ attached to s (see Section 3.4.3), $\det(\mathbf{H}^\mathrm{T}\mathbf{H})$ may become zero in two cases, according to Table 3.4:

1. $\wp(s)$ is *non-degenerate*, the set of measurements $\mathbf{\Psi^m}$ is *compatible* with $\wp(s)$, and we have $Rank(\mathbf{H}) = 2$. We call such s a *ghost solution*.

2. $\wp(s)$ is *degenerate*, and the set of measurements $\mathbf{\Psi^m}$ is *not compatible* with $\wp(s)$ where $Rank(\mathbf{H}) \leq 2$.

Therefore, we have the following proposition:

Proposition 18 Given a matrix $\mathbf{H}$ formed by the observations $\mathbf{\Psi^m}$ and parameterized by s, $\det(\mathbf{H}^\mathrm{T}\mathbf{H})$ vanishes at s if and only if s is i) the actual distance s^*; ii) a ghost solution; and iii) attached to a degenerate geometrical configuration.

In other words, a ghost solution s is a zero of the function $\det(\mathbf{H}^\mathrm{T}\mathbf{H})$ and arises when a non-degenerate geometrical configuration $\wp(s)$ yields reflected lines with orientations equivalent with those measured in the image plane. This leads to an ambiguity in our reconstruction scheme: any mirror surface whose location and shape is determined by a ghost solution would generate exactly the same measurements $\mathbf{\Psi^m}$ as the ones generated by the actual mirror surface (attached to actual solution s^*).

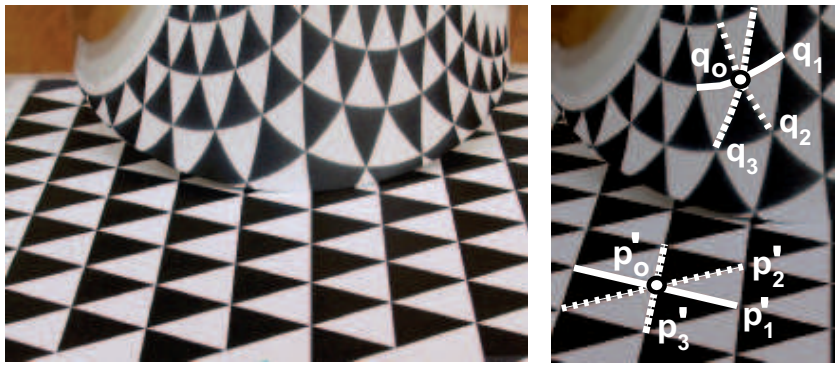


Figure 3.5: **Data gathered from the image**. In this example, the measurements are the orientations of the reflected curves $\mathbf{q}_1$, $\mathbf{q}_2$ and $\mathbf{q}_3$ evaluated at the intersecting point $\mathbf{q}_0$ and the position of $\mathbf{q}_0$ in the image plane.

3.5.3.2 Reconstruction Algorithm

To obtain three orientations at each examined point, we can adopt a pattern composed of a tessellation of black and white equilateral triangles. Edges of the pattern grids act as a triplet of intersecting lines and corners serve as mark points (see Figure 3.5). As indicated in Figure 3.5, the triplet of scene lines $\mathbf{p}'_1$, $\mathbf{p}'_2$, $\mathbf{p}'_3$ intersecting at $\mathbf{q}_0$ are reflected to $\mathbf{q}_1$, $\mathbf{q}_2$ and $\mathbf{q}_3$, respectively. Notice that $\mathbf{p}'_1$, $\mathbf{p}'_2$, $\mathbf{p}'_3$ are the images of the scene lines $\mathbf{p}_1$, $\mathbf{p}_2$, $\mathbf{p}_3$. Given a scene line $\mathbf{p}_i$, we may accurately measure the orientation of its reflected image curve $\mathbf{q}_i$ at $\mathbf{q}_0$ using B-spline interpolation. In fact, by constructing a B-spline that interpolates image points along the curve, the direction of $\dot{\mathbf{q}}$ (i.e., $\tan\phi$) can be calculated by numerically differentiating the resulting B-spline. Using these numerical estimates, our local surface reconstruction algorithm at each grid point of the reflected pattern can be summarized in Table 3.5.

3.5.3.3 Numerical Simulations and Discussion

In order to validate our theoretical results, we have implemented a program in MatLab to simulate specular reflections. Given a set of scene lines, a known surface, and the observer, the program computes the corresponding reflected curve imaged by the observer. Thus, all of the geometrical quantities needed to form the matrix $\mathbf{H}$ can be numerically obtained.

Fig. 3.6 shows the profile of $\det(\mathbf{H}^{\mathrm{T}}\mathbf{H})$ as a function of s for an instance of geometrical configuration. The function $\det(\mathbf{H}^{\mathrm{T}}\mathbf{H})$ vanishes at s^* (actual solution) and s' (degenerate geometrical configuration). According to Definition 3, we can analytically predict degenerate geometrical configurations by checking whether the value of s satisfies that $\forall i \neq j$, $\frac{B_{u_i}}{B_{v_i}} = \frac{B_{u_j}}{B_{v_j}}$. For instance, as in Fig. 3.6, since the surface point associated with s' belongs to the plane defined by the scene lines, we know from Proposition 11 that s' gives rise to a degenerate geometrical configuration and should be rejected. Additionally, empirical analysis with Matlab shows that the more one (or more) scene line approach the principal plane, the lower is the slope of $\det(\mathbf{H}^{\mathrm{T}}\mathbf{H})$

1. Select a checkerboard intersection point and its corresponding reflected point (e.g., points $\mathbf{p}'_0$ and $\mathbf{q}_0$ in Figure 3.5).
2. Select a triplet of lines from the checkerboard pattern (e.g., image lines $\mathbf{p}'_1$, $\mathbf{p}'_2$, $\mathbf{p}'_3$) and corresponding reflected triplet of curves (e.g., $\mathbf{q}_1$, $\mathbf{q}_2$ and $\mathbf{q}_3$).
3. Compute the position of $\mathbf{p}_0$ and directions $\Delta\mathbf{p}_1$, $\Delta\mathbf{p}_2$, $\Delta\mathbf{p}_3$ of $\mathbf{p}_1$ $\mathbf{p}_2$, $\mathbf{p}_3$, respectively at $\mathbf{p}_0$ by interpolating B-splines through each of $\mathbf{p}'_1$, $\mathbf{p}'_2$, $\mathbf{p}'_3$ and by numerical differentiation at $\mathbf{p}'_0$.
4. Estimate the position of $\mathbf{q}_0$ and directions $\tan\phi_1^m$, $\tan\phi_2^m$, $\tan\phi_3^m$ of $\mathbf{q}_1$, $\mathbf{q}_2$ and $\mathbf{q}_3$ respectively at $\mathbf{q}_0$ from B-spline numerical differentiation.
5. Compute matrix $\mathbf{H}$ (Eq. (3.74)) from $\mathbf{p}_0$, $\mathbf{q}_0$, $\Delta\mathbf{p}_1$, $\Delta\mathbf{p}_2$, $\Delta\mathbf{p}_3$, $\tan\phi_1^m$, $\tan\phi_2^m$, $\tan\phi_3^m$.
6. Recover the distance s from $\det(\mathbf{H}^T\mathbf{H})$ (up to ghost solutions).
7. Recover the reflection point $\mathbf{r}_0$ on the mirror surface and the surface normal vector at $\mathbf{r}_0$ by Eq. (3.1) and Eq. (3.5), respectively.
8. Recover curvature parameters a, b, c by Eq. (3.76) up to one unknown parameter.

Table 3.5: Algorithm A1

at s^*. This in turn leads to lowered accuracy in estimating s^* (e.g., see the point highlighted with dashed circle in Figure 3.15 for details).

In this instance of simulation, no ghost solutions arise. In general, the existence of ghost solutions is more problematic for our reconstruction because it cannot be ruled out analytically. However, extensive numerical experiments with MatLab show that ghost solutions are rather uncommon. Notice that the number n of rows of $\mathbf{H}$ is equal to the number of scene lines. Thus, the greater is n, the lower is the probability that there exists a value of $s \neq s^*$ such that the set of measurements $\mathbf{\Psi^m}$ is compatible with $\wp(s)$. Additionally, ghost solutions may be easily rejected if some form of prior global information on shape is available. Further work is needed in order to derive sufficient conditions to reject ghost solutions.

3.5.3.4 Recovering the Second-Order Parameters: the Second-Order Ambiguity

Once s is determined, we can recover the second-order surface parameters a, b, c as functions of one free parameter r from Eq. (3.76). As a consequence of Proposition 14, however, the free parameter r cannot be estimated from tangent orientations only. This leads to a fundamental ambiguity for the second-order surface parameters, as far as the image tangent orientations are concerned. We call this ambiguity *Second-Order Ambiguity*. We can interpret the ambiguity as follows: consider the family of geometrical configurations

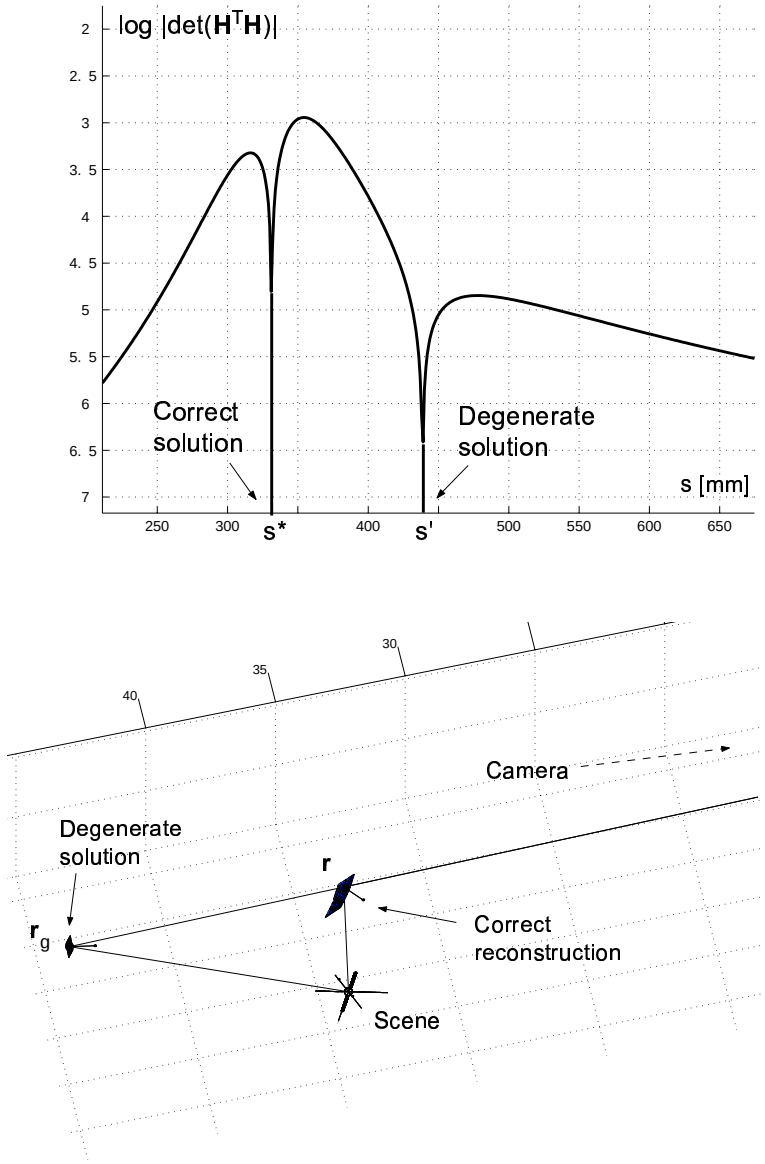
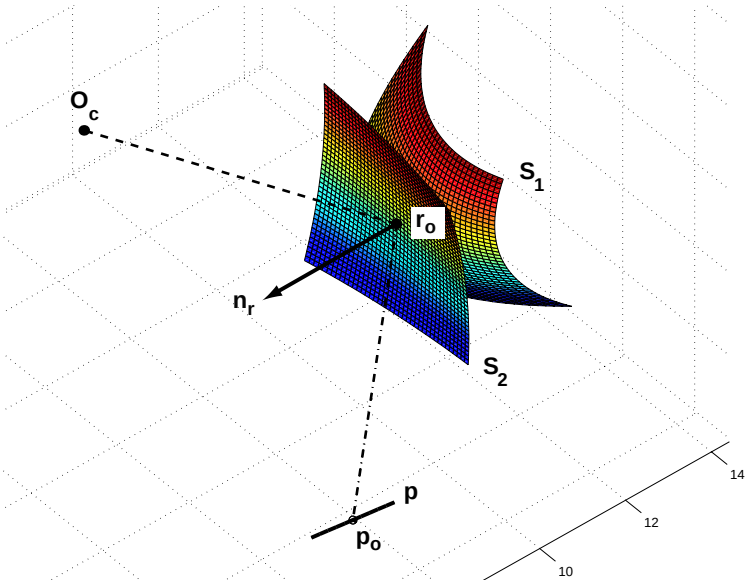


Figure 3.6: **Multiple solutions for** det($\mathbf{H}^{\mathrm{T}}\mathbf{H}$)**.** Top panel: profile of det($\mathbf{H}^{\mathrm{T}}\mathbf{H}$) as a function of s. The function det($\mathbf{H}^{\mathrm{T}}\mathbf{H}$) vanishes at $s = s^*$ (actual solution) and at $s = s'$ (solution attached to a degenerate geometrical configuration). Bottom panel: the corresponding reconstructed point for $s = s^*$ is the real surface point $\mathbf{r}$, and the reconstructed point for $s = s'$ is $\mathbf{r}_g$, which can easily be rejected as it belongs to the plane defined by the scene lines.

$\wp(r)$, where $\mathbf{c}$, $\mathbf{p}_0$, $\mathbf{p}_1$, $\mathbf{p}_2$, $\mathbf{r}_0(s)$ $\mathbf{n}_r(s)$ are fixed and a, b, c parameterized by r, according to Eq. (3.76). Then, given an arbitrary scene line passing through $\mathbf{p}_0$, all the mirror surfaces attached to the family will produce reflected image curves with invariant tangent orientation. In Fig. 3.7, an example of Second-Order Ambiguity is presented.



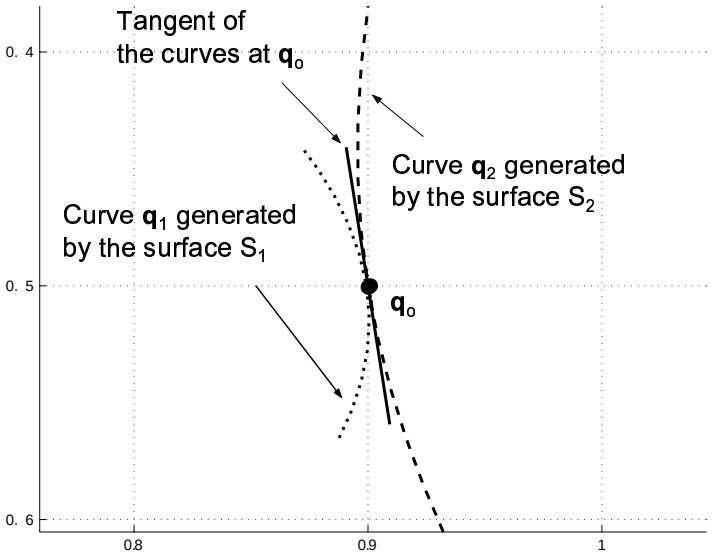


Figure 3.7: **Example of Second-Order Ambiguity.** Top panel: two mirror paraboloids S_1 and S_2 passing through $\mathbf{r}_0$, sharing the same orientation $\mathbf{n}_r$ at $\mathbf{r}_0$ and having the second-order surface parameters defined by Eq. (3.76) for two arbitrary values r_1 and r_2 of r. In this example, r_1 is chosen in order for S_1 to have elliptic curvature. r_2 is chosen in order for S_2 to have hyperbolic curvature. $\mathbf{p}$ is a scene line passing through $\mathbf{p}_0$. Bottom panel: $\mathbf{q}_1$ and $\mathbf{q}_2$ are the images of the reflected scene line $\mathbf{p}$ off the two mirrors S_1 and S_2, respectively. Notice that the tangents of the reflected curves at $\mathbf{q}_0$ are coincident as predicted by Eq. (3.76) and Eq. (3.65). We call this property Second-Order Ambiguity.

3.5.3.5 Recovering the Second-Order Parameters: Two Special Cases

In the following, we show how to resolve the Second-Order Ambiguity and recover the parameter r under two special cases:

Sphere. If the mirror surface is a sphere with radius R, then the second-order parameters satisfy $a = b$ and $c = 0$. By imposing $a = b$ to Eq. (3.76), we resolve the Second-Order Ambiguity and obtain

$$r = \frac{J_u - J_v}{h_1 - h_2}, \tag{3.77}$$

and the radius of the sphere being

$$R = \frac{2(h_1 - h_2)\cos\theta}{J_v h_1 - J_u h_2} \tag{3.78}$$

Additionally, by imposing that c = 0, we have $r\ h_3/\cos\theta$ = 0. Since $\cos\theta \neq 0$ and $r \neq 0$ (otherwise the corresponding geometrical configuration would be degenerate), h_3 must be zero. Namely:

$$-B_{v_i} B_{u_j} \tan\phi_j + B_{u_i} B_{v_j} \tan\phi_i = 0 \tag{3.79}$$

Notice that the quantity in Eq. (3.79) is the determinant of the 2×2 matrix obtained by taking the i^{th} and j^{th} rows and the first two columns of **H**. Eq. (3.79) gives a necessary condition for the mirror surface to be locally described as a spherical paraboloid around the reflection point, that is, Eq. (3.79) holds for any pair of scene lines.

Cylinder. Let us consider the circular section (with radius R) of a cylinder. The surface can be locally described as a parabolic paraboloid following the relationship $c^2 = ab$ (see [15]). Again from Eq. (3.76),

$$r = \frac{(J_v h_1 + J_u h_2) \pm \sqrt{(J_v h_1 + J_u h_2)^2 - 4(h_1 h_2 - h_3^2) J_u J_v}}{2(h_1 h_2 - h_3^2)} \tag{3.80}$$

It is not difficult to show that the radius of the circular section is $R = \frac{\cos^2\xi}{2a}$, with $\xi = \arctan(b/a)$. The Second-Order Ambiguity is solved up to a sign.

3.5.3.6 Recovering the Second-Order Parameters: General Case

In order to reconstruct the second-order parameters of a generic specular surface, we are required to determine the parameter r, which from Proposition 14, may not be solved by using more scene lines. A straightforward approach to determine r is to apply our second-order differential analysis and curvature measurement for additional constraints. Specifically, suppose that we can estimate the corresponding curvature κ_i^m of $\mathbf{q}_i(t)$ at

$\mathbf{q}_0$. We may equate this image curvature with its analytical expression κ_i^q derived in Eq. (3.57) , yielding

$$\kappa_i^m = \kappa_i^q = \frac{s\left\langle \mathbf{d}, \mathbf{v} \right\rangle^3}{l} \frac{\left\langle \ddot{\mathbf{r}}_i, \mathbf{v} \right\rangle \left[\dot{\mathbf{r}}_i,\, \mathbf{v}, \mathbf{d}\right] - \left\langle \dot{\mathbf{r}}_i, \mathbf{v} \right\rangle \left[\ddot{\mathbf{r}}_i,\, \mathbf{v},\, \mathbf{d}\right] + \left\langle \mathbf{d}, \mathbf{v} \right\rangle \left[\ddot{\mathbf{r}}_i,\, \mathbf{v},\, \dot{\mathbf{r}}_i\right]}{\left(\left|\left| \dot{\mathbf{r}}_i \right|\right|^2 \left\langle \mathbf{d}, \mathbf{v} \right\rangle^2 + \left\langle \dot{\mathbf{r}}_i, \mathbf{v} \right\rangle^2 - 2 \left\langle \dot{\mathbf{r}}_i, \mathbf{v} \right\rangle \left\langle \dot{\mathbf{r}}_i, \mathbf{d} \right\rangle \left\langle \mathbf{d}, \mathbf{v} \right\rangle \right)^{3/2}} \tag{3.81}$$

Unfortunately, since $\ddot{\mathbf{r}}_i$ derived in Eq. (3.37) is dependent on not only $s, r(a, b, c)$, but also the third-order surface parameters e, f, g, h, in general, the constraint imposed by Eq. (3.81) does not allow us to solve for r from curvature measurement κ_i^m. However, for a class of mirror surface whose third-order terms are negligible, we are able to recover the surface up to second-order accuracy using the constraint (3.81). As derived in Eq. (3.37), $\ddot{\mathbf{r}}_i$ can be divided into two parts: $\ddot{\mathbf{r}}_{i_1}$ and $\ddot{\mathbf{r}}_{i_2}$, where only the latter term depends on e, f, g, h. When $e, f, g, h \rightarrow 0$, $\ddot{\mathbf{r}}_i$ on the right-hand side of Eq. (3.81) simplifies to $\ddot{\mathbf{r}}_{i_1}$. Accordingly, the right-hand side of Eq. (3.81) becomes a function of s, r only. With s determined, we can estimate the free parameter r from the constraint (3.81) formed from the image curvature measurement of a single scene line, and then the second-order parameters a, b, c from Eq. (3.76). However, usage of three or more lines (combined with their corresponding curvature measurements) allows more robust estimation to reduce error from noise. For instance, we may estimate the parameter r by minimizing a mean curvature error function defined as follows:

$$E_\kappa(r) = \|\kappa^{\mathbf{m}} - \kappa^{\mathbf{q}}(\mathbf{r})\| \tag{3.82}$$

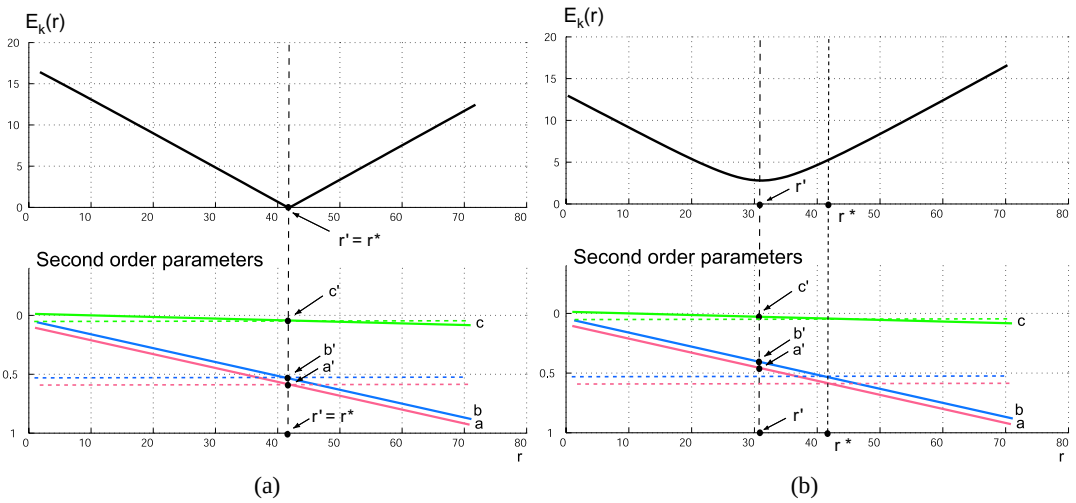
where $\kappa^{\mathbf{m}} = [\kappa_{\mathbf{1}}^{\mathbf{m}} \kappa_{\mathbf{2}}^{\mathbf{m}} \ldots \kappa_{\mathbf{n}}^{\mathbf{m}}]^{\mathbf{T}}$ and $\kappa^{\mathbf{q}} = [\kappa_{\mathbf{1}}^{\mathbf{q}} \kappa_{\mathbf{2}}^{\mathbf{q}} \ldots \kappa_{\mathbf{n}}^{\mathbf{q}}]^{\mathbf{T}}$ (see Fig. 3.8).

We conclude this section by showing that for large values of r, there exists a linear relationship between image curvature and r. This property may used to estimate r in close form solution (rather than by least square minimization of Eq. (3.82)).

Proposition 19 The image curvature κ for a scene line is asymptotically linearly dependent on the free parameter r.

Proof: We first express $\dot{\mathbf{r}}$ in terms of the parameter r. Rewriting Eq. (3.76), we get

$$\begin{cases} rh_1 &= J_u - 2a\cos\theta, \\ rh_2 &= J_v - 2b\cos\theta, \\ rh_3 &= 2c\cos\theta \end{cases} . \tag{3.83}$$



(a) (b)

Figure 3.8: (a) **Upper**: Mean curvature error function $E_\kappa(r)$ when the third-order parameters are negligible. The estimated r is the value that minimizes $E_\kappa(r)$. Since the third-order parameters are negligible, estimated (r') and actual (r^*) values of r are coincident. **Lower**: The 3 solid lines are the second-order parameters of the surface as a function of r (namely, $a(r), b(r), c(r)$). The dashed lines indicate the actual value of a, b, c. Solid and dashed lines intersect at $r' = r^*$. Thus, the estimated second-order surface parameters (a', b', c') are calculated with no error. (b) **Upper**: Mean curvature error function $E_\kappa(r)$ when the third-order parameters are not negligible. The estimated r is the value that minimizes $E_\kappa(r)$. Since the third-order parameters are not negligible, r' and r^* are no necessarily coincident. **Lower**: The estimated second-order surface parameters (a', b', c') are calculated at $r = r' \neq r^*$ and approximate the actual ones with non-zero error.

Using Eq. (3.83), we can express Δ in Eq. (3.19) as

$$\Delta \;=\; r^2(h_1h_2 - h_3^2). \tag{3.84}$$

It then follows from Eq. (3.9) that

$$\dot{\mathbf{r}} \;=\; -\frac{1}{r(h_1h_2 - h_3^2)} \begin{bmatrix} h_2 & h_3 & 0 \\ h_3 & h_1 & 0 \\ 0 & 0 & r(h_1h_2 - h_3^2) \end{bmatrix} \begin{bmatrix} B_u \\ B_v \\ 0 \end{bmatrix}. \tag{3.85}$$

As a result, we may express

$$\dot{\mathbf{r}} \;=\; \frac{1}{r}[q_1 \;\; q_2 \;\; 0], \tag{3.86}$$

where q_1, q_2 are independent of r. Furthermore, we may express $\dot{\lambda}(0)$ in terms of r using Eq. (3.9). That is,

$$\dot{\lambda}(0) \;=\; \frac{J_w(h_2B_u + h_3B_v)}{r(h_1h_2 - h_3^2)} - B_w \;=\; \frac{1}{r}\lambda_1 - B_w, \tag{3.87}$$

with λ_1 independent of r. We then express $\ddot{\mathbf{r}}$ in Eq. (3.38) and Eq. (3.40) in terms of r using Eq. (3.83) and Eq. (3.86). That is,

$$\begin{aligned}
\ddot{w}(0) &= \frac{1}{2r^2\cos\theta}\begin{bmatrix} q_1 & q_2 & 0 \end{bmatrix}\begin{bmatrix} J_u - rh_1 & rh_3 & 0 \\ rh_3 & J_v - rh_2 & 0 \\ 0 & 0 & 0 \end{bmatrix}\begin{bmatrix} q_1 \\ q_2 \\ 0 \end{bmatrix} \\
&= \frac{1}{2r^2\cos\theta}\left[(J_u q_1^2 + J_v q_2^2) - r(h_1 q_1^2 - 2h_3 q_1 q_2 + h_2 q_2^2)\right] \\
&= \frac{1}{r^2}l_0 + \frac{1}{r}l_1 \qquad (3.88)
\end{aligned}$$

with l_0, l_1 as terms independent of r, and

$$\begin{bmatrix} \ddot{u}(0) \\ \ddot{v}(0) \end{bmatrix} = -\frac{1}{r(h_1h_2 - h_3^2)}\begin{bmatrix} h_2 & h_3 \\ h_3 & h_1 \end{bmatrix}\begin{bmatrix} D_1 + D_1' + J_w\ddot{w}(0) \\ D_2 + D_2' \end{bmatrix}. \qquad (3.89)$$

Let us examine the dependence of D_1, D_2, D_1', D_2' on the parameter r. It follows from Eq. (3.46) that

$$\begin{aligned}
[D_1 \;\; D_2 \;\; D_3]^{\mathrm{T}} &= -\mathbf{A}\Delta\mathbf{p} + 2\mathbf{A}\,\dot{\mathbf{r}} + \dot{\mathbf{r}}^{\mathrm{T}}\tilde{\mathbf{C}}\,\dot{\mathbf{r}} + 2\dot{\lambda}(0)\mathbf{H}_g\dot{\mathbf{r}}), \\
[D_1' \;\; D_2' \;\; D_3']^{\mathrm{T}} &= \lambda\dot{\mathbf{r}}^{\mathrm{T}}\tilde{\mathbf{C}}'\,\dot{\mathbf{r}} \qquad (3.90)
\end{aligned}$$

where the tensors[4] $\tilde{\mathbf{C}}$ and $\tilde{\mathbf{C}}'$ are independent of r, derived from Eq. (3.46) as

$$\tilde{\mathbf{C}} = [\mathbf{C}_r^1 + \mathbf{C}_i^1 \;\; \mathbf{C}_r^2 + \mathbf{C}_i^2 \;\; \mathbf{C}_r^3 + \mathbf{C}_i^3], \quad \tilde{\mathbf{C}}' = [\mathbf{C}_h^1 \;\; \mathbf{C}_h^2 \;\; \mathbf{C}_h^3].$$

Let $\mathbf{v}_i$ denote vectors independent of the parameter r. We may rewrite each term in Eq. (3.90) as

$$\begin{aligned}
-\mathbf{A}\Delta\mathbf{p} &= \mathbf{v}_0, \quad 2\mathbf{A}\,\dot{\mathbf{r}} = \frac{1}{r}\mathbf{v}_1, \quad \dot{\mathbf{r}}^{\mathrm{T}}\tilde{\mathbf{C}}\,\dot{\mathbf{r}} = \frac{1}{r^2}\mathbf{v}_2, \quad \dot{\mathbf{r}}^{\mathrm{T}}\tilde{\mathbf{C}}'\,\dot{\mathbf{r}} = \frac{1}{r^2}\mathbf{v}_3, \\
\dot{\lambda}(0)\mathbf{H}_g\dot{\mathbf{r}} &= -\frac{\lambda_1/r - B_w}{2r\cos\theta}\begin{bmatrix} J_u - rh_1 & rh_3 & 0 \\ rh_3 & J_v - rh_2 & 0 \\ 0 & 0 & 0 \end{bmatrix}\begin{bmatrix} q_1 \\ q_2 \\ 0 \end{bmatrix} \\
&= \frac{1}{r^2}\mathbf{v}_4 + \frac{1}{r}\mathbf{v}_5 + \mathbf{v}_6.
\end{aligned}$$

Therefore, D_1, D_2, D_1', D_2' are linear combinations of $\left\{\frac{1}{r^2}, \frac{1}{r}, r^0\right\}$. It then follows from Eq. (3.89) and Eq. (3.88) that $\ddot{\mathbf{r}}$ depends linearly on $\left\{\frac{1}{r^3}, \frac{1}{r^2}, \frac{1}{r}\right\}$. Consequently, it is obvious from Eq. (3.57) that the image

[4] Each column of these matrix-format tensors is another matrix.

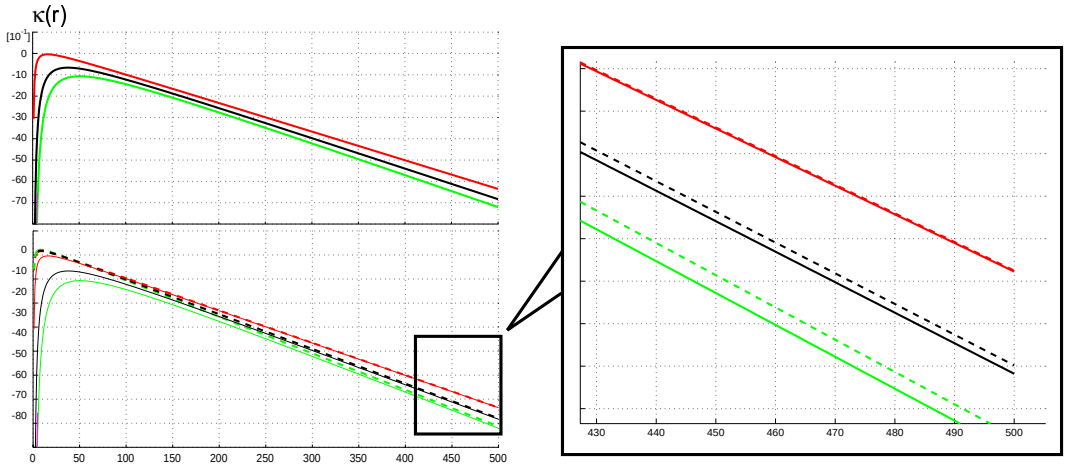


Figure 3.9: Numerical validation of Proposition 19. **Upper left panel**: Curvatures of three reflected image curves (depicted with different color codes) are plotted as functions of the parameter r when the third-order parameters are not necessarily negligible. Notice the linear behavior of the curvatures as r approaches infinity. In the **lower left panel** the plots shown in the upper left panel are compared with those obtained when the third-order surface parameters are negligible (dashed bold lines). Notice that the profiles (both asymptotically linear) have the same slope as predicted by our theoretical analysis (see zoom-in box in the **right panel**).

curvature κ for a scene line actually depends linearly on $\left\{\frac{1}{r}, r^0, r^1\right\}$, or equivalently, κ depends asymptotically linearly on r. ■

As a result, we may express the image curvature as

$$\kappa = \kappa_0 \frac{1}{r} + \kappa_1 + \kappa_2 r, \tag{3.91}$$

where expression for κ_0, κ_1 and κ_2 have been derived in the proof of Proposition 19. It follows that the third-order surface parameters affect coefficients κ_0 and κ_1 only, which is confirmed by numerical simulation in Fig. 3.9.

We can apply Eq. (3.91) to obtain an approximated closed form solution for r when the third-order parameters are negligible and $r \gg 0$. If the image curvature measurements of $N (N \geq 1)$ lines are available,

$$r \simeq \frac{1}{N} \sum_{i}^{N} \frac{(\kappa)_i^m - (\kappa_1)_i}{(\kappa_2)_i} \tag{3.92}$$

where quantities attached to the i^{th} curve are indicated with the subscript i. As a result, r and the second-order surface parameters (a, b, c) can be estimated in closed form solution.

3.5.3.7 Recovering the Third-Order Parameters

As discussed in Section 3.3.2.3, the image curvatures depend linearly on the third-order surface parameters (see Proposition 8). Thus, curvature measurements can be further employed to constraint the third-order surface parameters. In fact, suppose that $N \geq 4$ scene lines are considered and that the corresponding measurements of curvature are available in the image plane. We can derive from Eq. (3.57) the following system (in matrix form):

$$\kappa^{\mathbf{m}} = \mathbf{k} + \mathbf{K}\,[\mathbf{e}\ \mathbf{f}\ \mathbf{g}\ \mathbf{h}]^{\mathbf{T}} \tag{3.93}$$

where $\kappa^{\mathbf{m}} = [\kappa_{\mathbf{1}}^{\mathbf{m}} \kappa_{\mathbf{2}}^{\mathbf{m}} \ldots \kappa_{\mathbf{n}}^{\mathbf{m}}]^{\mathbf{T}}$ is the vector of measured curvatures, and the vector $\mathbf{k}$ and $N \times 4$ matrix $\mathbf{K}$ are function of r and may be derived from Eq. (3.57). Numerical analysis shows that $\mathbf{K}$ is, in general, non-singular. Thus, the third-order surface parameters can be expressed as functions of the free parameter r.

One may wonder whether by adding (at least) a fifth additional curvature measurement, the free parameter r can be found. By adding (at least) a fifth additional curvature measurement, exploiting the asymptotic linearity of κ and the fact that κ_1 does not depend on r, we have

$$\kappa^{\mathbf{m}} \simeq \mathbf{k}' + \mathbf{K}'\,[\mathbf{e}\ \mathbf{f}\ \mathbf{g}\ \mathbf{h}\ \mathbf{r}]^{\mathbf{T}} \tag{3.94}$$

where vector $\mathbf{k}'$ and $N \times 5$ matrix $\mathbf{K}'$ can be derived from Eq. (3.93) and Eq. (3.91); $N \geq 5$; $r \gg 0$. Numerical analysis shows that matrix $\mathbf{K}'$ is singular. Theoretical proof of this statement is postponed to future work. We speculate that the fifth (and additional) measurement would not carry independent information. As a result, likewise the second-order surface parameters, the third-order parameters would only be recovered up to the unknown r. We call this conjecture the *Third-Order Ambiguity*. According to this conjecture, any mirror paraboloid passing through a point $\mathbf{r}_0$, sharing the same orientation $\mathbf{n}_r$ at $\mathbf{r}_0$ and having their second and third-order surface parameters belonging to the family defined in Eq. (3.76) and Eq. (3.93) respectively, would yield reflected image curves with invariant orientation and curvature at $\mathbf{q}_0$.

3.5.4 Image Measurement: Orientations + Local Scale

In this section, we show that by measuring local position, orientation, and local scale in the neighborhood of a reflected point $\mathbf{q}_0$, the local surface geometry can be recovered up to third-order accuracy. Local scale information indicates to what degree points along a reflected curve are deformed (stretched or compressed) by the specular surface. Orientation and local scale information may be captured by measuring the full first-order

derivative of the reflected curves $\mathbf{q}_1(t)$, $\mathbf{q}_2(t) \ldots \mathbf{q}_n(t)$ at the point of intersection $\mathbf{q}_0$.

In Section 3.5.4.1, we show that by measuring the first-order derivatives of at least two reflected curves at $\mathbf{q}_0$, it is possible to estimate surface position ($\mathbf{r}_0$) and surface orientation at $\mathbf{r}_0$. Additionally, the second surface parameters (a, b, c) can be estimated in closed form solution up to a sign. In Section 3.5.4.2, we show that by measuring the second-order derivatives of at least two reflected curves at $\mathbf{q}_0$, the third-order surface parameters (e, f, g, h) can be estimated in closed form solution. In Section 3.5.4.3, we show how to measure first and second derivatives of the reflected curves. Given these measurements, we present an explicit reconstruction algorithm which may be summarized in eight steps. In Section 3.5.4.4, we discuss to what extent the truncation error due to finite approximation of the measurements affects the reconstruction accuracy. Finally, in Section 3.5.4.5, we show that the hypothesis on the structure of the scene (i.e., scene composed of a grid of intersecting lines) is, in fact, not necessary. Scale and orientation measurements can also be extracted from the reflection of a planar scene patch of arbitrary geometry. We show that local surface shape can be estimated if the location of at least three arbitrary points is available within a neighborhood of the reflected scene patch.

3.5.4.1 Recovering the First- and Second-Order Parameters

In Section 3.5.2, we showed that if we are able to measure the tangent directions ($\tan\phi_k$ and $\tan\phi_j$) for (at least) two image curves $\mathbf{q}_k(t)$ and $\mathbf{q}_j(t)$ at $\mathbf{q}_0$, the first- and second-order surface unknown parameters are reduced from s, a, b, c to r, s. We start our analysis from this result by defining

$$\mathbf{V} \;=\; \frac{1}{h_1 h_2 - h_3^2} \begin{bmatrix} h_2 & h_3 \\ h_3 & h_1 \end{bmatrix}, \qquad \mathbf{B} \;=\; \begin{bmatrix} B_u \\ B_v \end{bmatrix}.$$

We may re-express $\dot{\mathbf{r}}$ in Eq. (3.85) as

$$\dot{\mathbf{r}} \;=\; -\frac{1}{r} \begin{bmatrix} \mathbf{V}\,\mathbf{B} \\ 0 \end{bmatrix}. \tag{3.95}$$

Accordingly, it follows from Eq. (3.35) that the first-order derivative $\dot{\mathbf{q}}$ of the observed reflection curve in the image plane can also be expressed in terms of the two unknowns, r and s:

$$\dot{\mathbf{q}} \;=\; -\frac{1}{r}\mathbf{T} \begin{bmatrix} \mathbf{V}\,\mathbf{B} \\ 0 \end{bmatrix}, \tag{3.96}$$

where only the unknown s (not r) appears in $\mathbf{T}$, $\mathbf{V}$ and $\mathbf{B}$. Then, we can compute the L_2 norm of $\dot{\mathbf{q}}$ from Eq. (3.96) as

$$|| \dot{\mathbf{q}} ||^2 \;=\; \langle \dot{\mathbf{q}}, \dot{\mathbf{q}} \rangle \;=\; \frac{1}{r^2} \begin{bmatrix} \mathbf{B}^{\mathrm{T}}\mathbf{V}^{\mathrm{T}} & 0 \end{bmatrix} \mathbf{T}^{\mathrm{T}}\mathbf{T} \begin{bmatrix} \mathbf{VB} \\ 0 \end{bmatrix}. \tag{3.97}$$

Assume that we are able to estimate the first-order derivatives of $\mathbf{q}_k(t)$ and $\mathbf{q}_j(t)$ at $\mathbf{q}_0$ (see Section 3.5.4.3 for details). We denote them by $\dot{\mathbf{q}}_k^m$ and $\dot{\mathbf{q}}_j^m$, respectively. By taking the ratio $|| \dot{\mathbf{q}}_k^m ||^2 \,/\, || \dot{\mathbf{q}}_j^m ||^2$, we have

$$\frac{|| \dot{\mathbf{q}}_k^m ||^2}{|| \dot{\mathbf{q}}_j^m ||^2} \;=\; \frac{\begin{bmatrix} \mathbf{B}_k{}^{\mathrm{T}}\mathbf{V}^{\mathrm{T}} & 0 \end{bmatrix} \mathbf{T}^{\mathrm{T}}\mathbf{T} \begin{bmatrix} \mathbf{VB}_k \\ 0 \end{bmatrix}}{\begin{bmatrix} \mathbf{B}_j{}^{\mathrm{T}}\mathbf{V}^{\mathrm{T}} & 0 \end{bmatrix} \mathbf{T}^{\mathrm{T}}\mathbf{T} \begin{bmatrix} \mathbf{VB}_j \\ 0 \end{bmatrix}}, \tag{3.98}$$

where the matrix $\mathbf{V}$ is expressed in terms of our tangent direction measurements, $\tan\phi_k$ and $\tan\phi_j$. Notice that the matrix $\mathbf{T}$ defined in Eq. (3.29) does not depend on a particular line. Eq. (3.98) imposes a constraint for us to solve for s. Eq. (3.98) may have multiple solutions which we call *ghost solutions* as discussed in Section 3.5.3.1. Notice that the values of s for which the corresponding geometrical configuration is degenerate make the matrix V undefined. For those values of s, Eq. (3.98) cannot be computed.

Once s is computed, we can derive the closed form solution for another unknown r up to a sign from Eq. (3.97):

$$r^2 \;=\; \frac{\begin{bmatrix} \mathbf{B}_k{}^{\mathrm{T}}\mathbf{V}^{\mathrm{T}} & 0 \end{bmatrix} \mathbf{T}^{\mathrm{T}}\mathbf{T} \begin{bmatrix} \mathbf{VB}_k \\ 0 \end{bmatrix}}{|| \dot{\mathbf{q}}_k^m ||^2}. \tag{3.99}$$

3.5.4.2 Recovering the Third-Order Parameters

To recover the third-order surface parameters, we assume that we are able to estimate the second-order derivatives for the two reflection curves in the image plane, denoted by $\ddot{\mathbf{q}}_k^m (= [\ddot{u}_k^m \; \ddot{v}_k^m \; \ddot{w}_k^m])$ and $\ddot{\mathbf{q}}_j^m (= [\ddot{u}_j^m \; \ddot{v}_j^m \; \ddot{w}_j^m])$ (see Section 3.5.4.3 for details). In accordance with the decomposition of $\ddot{\mathbf{q}}$, we define

$$(\ddot{\mathbf{q}}_2)_k^m \;=\; \ddot{\mathbf{q}}_k^m - (\ddot{\mathbf{q}}_1)_k, \quad (\ddot{\mathbf{q}}_2)_j^m \;=\; \ddot{\mathbf{q}}_j^m - (\ddot{\mathbf{q}}_1)_j, \tag{3.100}$$

where $(\ddot{\mathbf{q}}_1)_k$ and $(\ddot{\mathbf{q}}_1)_j$ are known from Eq. (3.38), once we have recovered s, a, b, c. Letting $\mathbf{T}_{22}$ denote the upper left 2×2 sub-matrix of $\mathbf{T}$, and letting $\hat{\ddot{\mathbf{q}}}_2$ denote a vector consisting of the first two components of $\ddot{\mathbf{q}}_2$, it follows from Eq. (3.56) that

$$\hat{\ddot{\mathbf{q}}}_2 \;=\; \mathbf{T}_{22} \begin{bmatrix} \ddot{u}_2 \\ \ddot{v}_2 \end{bmatrix}. \tag{3.101}$$

Similar to the first-order analysis, we may re-express Eq. (3.40) as

$$\begin{bmatrix} \ddot{u}_2 \\ \ddot{v}_2 \end{bmatrix} \;=\; \frac{2\cos\theta}{r(h_1h_2 - h_3^2)} \begin{bmatrix} h_2 & h_3 \\ h_3 & h_1 \end{bmatrix} \begin{bmatrix} \dot{u}^2 & 2\dot{u}\dot{v} & \dot{v}^2 & 0 \\ 0 & \dot{u}^2 & 2\dot{u}\dot{v} & \dot{v}^2 \end{bmatrix} \begin{bmatrix} e \\ f \\ g \\ h \end{bmatrix}. \tag{3.102}$$

Letting $(\hat{\ddot{\mathbf{q}}}_2)_k^m$ and $(\hat{\ddot{\mathbf{q}}}_2)_j^m$ denote vectors consisting of the first two components of $(\ddot{\mathbf{q}}_2)_k^m$ and $(\ddot{\mathbf{q}}_2)_j^m$, and using Eq. (3.101) and Eq. (3.102), we get a constraint system for e, f, g, h by equating the estimates $(\hat{\ddot{\mathbf{q}}}_2)_k^m$ and $(\hat{\ddot{\mathbf{q}}}_2)_j^m$ with their real values. That is

$$\begin{bmatrix} (\hat{\ddot{\mathbf{q}}}_2)_k^m \\ (\hat{\ddot{\mathbf{q}}}_2)_j^m \end{bmatrix} \;=\; \frac{2\cos\theta}{r(h_1h_2 - h_3^2)} \mathbf{M}_1\mathbf{M}_2\mathbf{M}_3 \begin{bmatrix} e \\ f \\ g \\ h \end{bmatrix}, \tag{3.103}$$

where $\mathbf{M}_1, \mathbf{M}_2, \mathbf{M}_3$ are defined as follows:

$$\mathbf{M}_1 \;=\; \begin{bmatrix} h_2 & h_3 & 0 & 0 \\ h_3 & h_1 & 0 & 0 \\ 0 & 0 & h_2 & h_3 \\ 0 & 0 & h_3 & h_1 \end{bmatrix}, \; \mathbf{M}_2 \;=\; \begin{bmatrix} \mathbf{T}_{22} & \mathbf{0} \\ \mathbf{0} & \mathbf{T}_{22} \end{bmatrix},$$

$$\mathbf{M}_3 \;=\; \begin{bmatrix} \dot{u}_k^2 & 2\dot{u}_k\dot{v}_k & \dot{v}_k^2 & 0 \\ 0 & \dot{u}_k^2 & 2\dot{u}_k\dot{v}_k & \dot{v}_k^2 \\ \dot{u}_j^2 & 2\dot{u}_j\dot{v}_j & \dot{v}_j^2 & 0 \\ 0 & \dot{u}_j^2 & 2\dot{u}_j\dot{v}_j & \dot{v}_j^2 \end{bmatrix}. \tag{3.104}$$

Suppose that we observe two curves $\mathbf{q}_k(t)$ and $\mathbf{q}_j(t)$ with different orientations at the point $\mathbf{q}_0$. Eq. (3.103) leads to the following closed-form solution for the third-order surface parameters, that is,

$$\begin{bmatrix} e \\ f \\ g \\ h \end{bmatrix} = \frac{r(h_1 h_2 - h_3^2)}{2\cos\theta} (\mathbf{M}_1\mathbf{M}_2\mathbf{M}_3)^{-1} \begin{bmatrix} (\hat{\ddot{\mathbf{q}}}_2)_k^m \\ (\hat{\ddot{\mathbf{q}}}_2)_j^m \end{bmatrix}, \tag{3.105}$$

where $(\hat{\ddot{\mathbf{q}}}_2)_k^m$, $(\hat{\ddot{\mathbf{q}}}_2)_j^m$ denote the first two components of $(\ddot{\mathbf{q}}_2)_k^m$, $(\ddot{\mathbf{q}}_2)_j^m$, as defined in Eq. (3.100).

The existence of $(\mathbf{M}_1\mathbf{M}_2\mathbf{M}_3)^{-1}$ is based on the following proposition:

Proposition 20 The matrix $\mathbf{M}_1\mathbf{M}_2\mathbf{M}_3$ is invertible.

Proof: $\det(\mathbf{M}_1) \neq 0$ follows directly from $\Delta \neq 0$. Let $\mathbf{d} = (\mathbf{d}_x, \mathbf{d}_y, \mathbf{d}_z)$ in the reference system $[\mathbf{x}\,\mathbf{y}\,\mathbf{z}]$ centered at $\mathbf{c}$, with $\mathbf{v} = (0, 0, -1)$. We can express $\mathbf{T}$ in Eq. (3.29) as

$$\mathbf{T} = \frac{l}{s\mathbf{d}_z} \begin{bmatrix} 1 & 0 & -\mathbf{d}_x/\mathbf{d}_z \\ 0 & 1 & -\mathbf{d}_y/\mathbf{d}_z \\ 0 & 0 & 0 \end{bmatrix}, \tag{3.106}$$

which is singular. Notice that $\mathbf{T}_{22}$ is non-singular. Accordingly, $\det(\mathbf{M}_2) \neq 0$. Therefore, we only need to prove that $\mathbf{M}_3$ is invertible.

We first show that $\mathbf{M}_3$ is invertible when one of $\{\dot{u}_k, \dot{v}_k, \dot{u}_j, \dot{v}_j\}$ is zero. For example, if $\dot{u}_k = 0$, then $\det(\mathbf{M}_3) = (\dot{v}_k \dot{u}_j)^4 \neq 0$. Otherwise, either $\dot{v}_k = 0$ or $\dot{u}_j = 0$ will contradict our observation of two curves with different orientations. Next, we shall consider the case where none of $\dot{u}_k, \dot{v}_k, \dot{u}_j, \dot{v}_j$ is zero. The proof is performed by contradiction. Under the assumption of observing two differently-oriented image curves, we should have

$$\dot{v}_k/\dot{u}_k \neq \dot{v}_j/\dot{u}_j, \quad \dot{v}_k/\dot{u}_k \neq -\dot{v}_j/\dot{u}_j. \tag{3.107}$$

Suppose that $\mathbf{M}_3$ is singular. Its four row vectors U_1, U_2, U_3, U_4 are linearly dependent. Without loss of generality, we may assume that $U_4 = k_1 U_1 + k_2 U_2 + k_3 U_3$ (at least one k_i is non-zero), which can be

expanded as

$$\begin{aligned} k_1\dot{u}_k^2 + k_3\dot{u}_j^2 &= 0 && (3.108)\\ 2k_1\dot{u}_k\dot{v}_k + k_2\dot{u}_k^2 + 2\,k_3\dot{u}_j\dot{v}_j &= \dot{u}_j^2 && (3.109)\\ k_1\dot{v}_k^2 + 2\,k_2\dot{u}_k\dot{v}_k + k_3\dot{v}_j^2 &= 2\dot{u}_j\dot{v}_j && (3.110)\\ k_2\dot{v}_k^2 &= \dot{v}_j^2 && (3.111) \end{aligned}$$

By eliminating $\dot{u}_j^2, \dot{u}_j\dot{v}_j$ in Eq. (3.109) through substitutions from equations (3.108), (3.110) and (3.111), we get

$$(k_1 + k_2k_3)(k_3\dot{v}_k + \dot{u}_k)^2 = 0. \tag{3.112}$$

Similarly, by eliminating $\dot{u}_k\dot{v}_k, \dot{v}_k^2$ in Eq. (3.110), we get

$$(k_1 + k_2k_3)(k_1\dot{v}_j - k_2\dot{u}_j)^2 = 0. \tag{3.113}$$

If $k_1 + k_2k_3 = 0$, then it follows from equations (3.108) and (3.111) that $k_3\left[\dot{v}_j^2/\dot{v}_k^2 - \dot{u}_j^2/\dot{u}_k^2\right] = 0$. To satisfy Eq. (3.107), we must have $k_3 = 0$, which leads to $k_1 = 0$ and then $\dot{v}_k/\dot{u}_k = \dot{v}_j/\dot{u}_j$, contradictory to our assumption (3.107). Consequently, the two constraints (3.112) and (3.113) become

$$k_3\dot{v}_k + \dot{u}_k = 0, \quad k_1\dot{v}_j - k_2\dot{u}_j = 0. \tag{3.114}$$

Equations (3.114), (3.108) and (3.111) present an over-constrained system for k_1, k_2, k_3. It follows that to satisfy all of them, we must have $\dot{v}_k/\dot{u}_k = \dot{v}_j/\dot{u}_j$, which is again, contradictory to our assumption (3.107). Therefore, U_1, U_2, U_3, U_4 must be linearly independent, and $\mathbf{M}_3$ is invertible. ■

3.5.4.3 Reconstruction Procedure

To obtain two orientations at each examined point, we adopt a checkerboard pattern of $2cm \times 2cm$ grid size. The edges of the pattern grids act as a pair of intersecting lines and corners serve as mark points (which provide scale information, see Figure 3.10). In Section 3.5.4.1 and Section 3.5.4.2, we have assumed that for a reflected curve $\mathbf{q}(t)$ observed on the image plane, we are able to measure its orientation $\tan\phi$, the first-order derivative $\dot{\mathbf{q}}^m$, and the second-order derivative $\ddot{\mathbf{q}}^m$ at $\mathbf{q}_0$. Next, we shall describe how to compute these

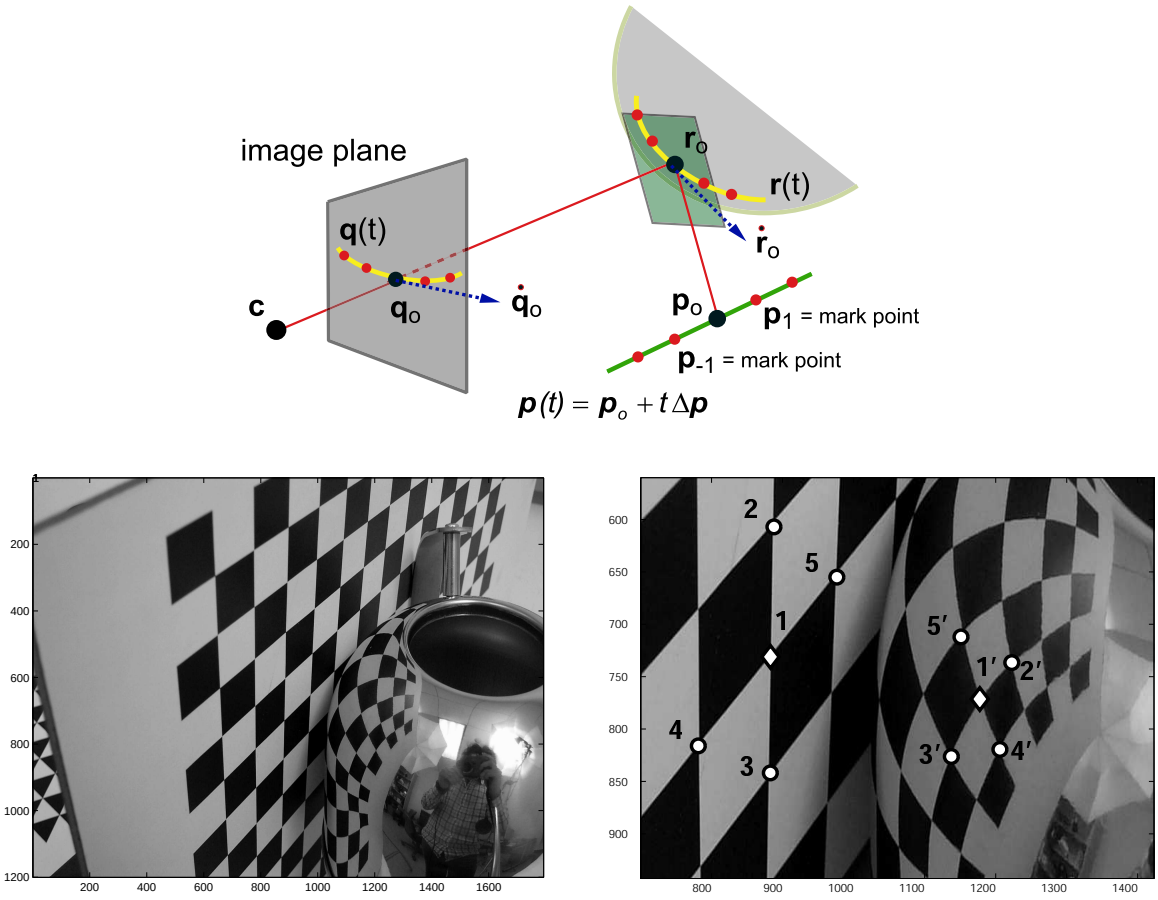


Figure 3.10: **Data gathered from the image.** In this example, the measurements are the orientations and scale along two lines passing through $1'$ and may be estimated from the position of the points $2'$, $3'$, $4'$, $5'$.

quantities numerically by combining B-spline interpolation and finite difference approximation.

Given a scene line $\mathbf{p}(t)$, we may accurately measure the orientation of its reflected image curve $\mathbf{q}(t)$ at $\mathbf{q}_0$ using B-spline interpolation. In fact, by constructing a B-spline that interpolates image points along the curve, the direction of $\dot{\mathbf{q}}$ (i.e., $\tan\phi$) can be calculated by numerical differentiation of the resulting B-spline. To estimate a complete $\dot{\mathbf{q}}$ (with both direction and magnitude) and higher-order derivative $\ddot{\mathbf{q}}$, we can make use of mark points $\mathbf{p}_0 = \mathbf{p}(t_0), \mathbf{p}_{-1} = \mathbf{p}(t_{-1}), \mathbf{p}_1 = \mathbf{p}(t_1), \ldots$ (see Figure 3.10) distributed along $\mathbf{p}(t)$, and use central finite difference approximation. Suppose that the mark points $\mathbf{p}(t_i)(i = \ldots, -1, 0, 1, \ldots)$ are mapped to corresponding image points $\mathbf{q}(t_i)$. Let the step size $\Delta t = t_i - t_{i-1}$. We may approximate $\dot{\mathbf{q}}$ and $\ddot{\mathbf{q}}$ at $\mathbf{q}_0$ by using two points and three points respectively. That is,

$$\dot{\mathbf{q}} \approx (\mathbf{q}(t_1) - \mathbf{q}(t_{-1}))/(2\Delta t) \tag{3.115}$$

$$\ddot{\mathbf{q}} \approx (\mathbf{q}(t_1) - 2\mathbf{q}(t_0) + \mathbf{q}(t_{-1}))/(\Delta t)^2. \tag{3.116}$$

Using these numerical estimates, our local surface reconstruction algorithm at the intersection points of the reflected pattern can be summarized in Table 3.6

1. Select a checkerboard intersection point and its reflected point (e.g., points 1 and $1'$ in Figure 3.10).
2. Select four neighboring points from both checkerboard pattern (e.g., 2, 3, 4, 5) and corresponding reflected pattern (e.g., $2'$, $3'$, $4'$, $5'$).
3. From 1, 2, 3, 4, 5 compute $\mathbf{p}_0$ and the direction of two scene lines $\Delta\mathbf{p}_1$ and $\Delta\mathbf{p}_2$.
4. From $1'$, $2'$, $3'$, $4'$, $5'$ estimate $\mathbf{q}^m$, $\dot{\mathbf{q}}_1^m$, $\dot{\mathbf{q}}_2^m$ and $\ddot{\mathbf{q}}_1^m$, $\ddot{\mathbf{q}}_2^m$ using equations (3.115) and (3.116).
5. Recover the distance s by Eq. (3.98) from $\dot{\mathbf{q}}_1^m$, $\dot{\mathbf{q}}_2^m$.
6. Recover the parameter r by Eq. (3.99) from $\dot{\mathbf{q}}_1^m$, $\dot{\mathbf{q}}_2^m$.
7. Recover curvature parameters (a,b,c) by Eq. (3.76).
8. Recover the third-order surface parameters (e,f,g,h) by Eq. (3.105) from $\ddot{\mathbf{q}}_1^m$, $\ddot{\mathbf{q}}_2^m$.

Table 3.6: Algorithm A2

3.5.4.4 Reconstruction Error

The reconstruction error of our algorithm mainly comes from the truncation error of the finite difference approximation (3.115). The truncation error decays when Δt decreases. To analyze how this numerical approximation affects the recovery of the distance, the curvature, and the third-order parameters of the mirror surface, we conducted numerical simulations on a synthetic mirror surface by implementing equations (3.98), (3.99) and (3.105) in Matlab. Given the center of the camera $\mathbf{c}$ and two scene lines intersecting at $\mathbf{p}_0$, we observe two reflected image curves depicted in Figure 3.12 (first panel). The synthetic mirror surface is positioned such that the distance s between the reflecting point $\mathbf{r}_0$ and $\mathbf{c}$ is 9cm. The surface principal curvatures at $\mathbf{r}_0$ are $\kappa_1 = -0.603$ and $\kappa_2 = -0.502$, and the third-order surface parameters are $e = -0.35$, $f = -0.1$, $g = 0.2$, and $h = -0.045$. By numerically measuring the first- and second-order derivatives at $\mathbf{q}_0$ (i.e., point 1) using pairs of mark points located at increasing distance Δt from $\mathbf{q}$ (i.e., mark point pair $(2, 2')$, $\cdots$ $(5, 5')$), we recover the local surface at $\mathbf{r}_0$ as described in Section 3.5.4.1 and Section 3.5.4.2. The remaining top and middle panels of Figure 3.12 plot each recovered surface parameter as a function of the mark gap Δt, with the maximum percentage error reported at the bottom. Notice that the error of recovered distance s increases as a quadratic function of Δt, the curvature error is one order of magnitude larger than

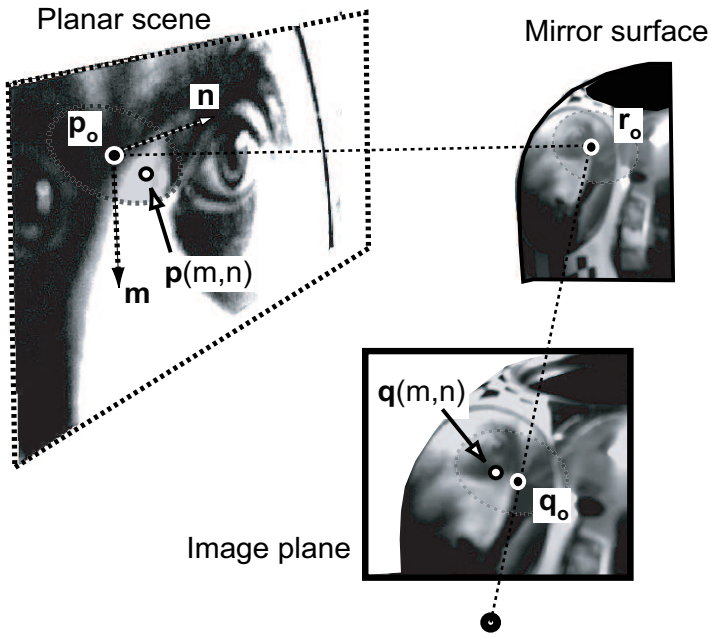


Figure 3.11: **Example of generalized mapping.** A set of points in the neighborhood of $\mathbf{p}_o$ (within a planar scene) are reflected off the mirror surface and observed in the image plane. By measuring the position of such reflected points in the image plane, it is possible to estimate the local shape of the surface around $\mathbf{r}_o$.

the distance error, and the third-order parameter error is one order of magnitude larger than the curvature error. In the bottom panel the reconstructed surface (estimated by using mark points $(5, 5')$) is qualitatively compared to the original one in both $\mathbf{w} - \mathbf{v}$ and $\mathbf{w} - \mathbf{u}$ sections of the $\mathbf{uvw}$ reference system. Numerical approximation (3.115) and (3.116) gives rise to reasonably good reconstruction results, as long as the mark points are close enough to each other (i.e., Δt small enough).

3.5.4.5 Generalized Mapping

In this section, we show that the hypothesis on the structure of the scene (i.e., scene composed of a grid of intersecting lines) is, in fact, not necessary. Information equivalent to scale and orientation measurements can also be extracted from the reflection of a planar scene patch of arbitrary geometry.

Let $\mathbf{p}_o$ be a scene point belonging to a planar scene patch. Let $\mathbf{r}_o$ be the corresponding reflection on the mirror, and $\mathbf{q}_o$ be its observation (see Figure 3.11). Set a reference system whose origin is located at $\mathbf{p}_o$ and whose orthogonal axis $\mathbf{m}$ and $\mathbf{n}$ belong to the scene patch and are arbitrarily orientated. Let m and n be the coordinates in the $[\mathbf{m}\ \mathbf{n}]$ reference system of a generic point $\mathbf{p}(m, n)$ belonging to the scene patch in a neighborhood of $\mathbf{p}_o$. Thus, we can define a mapping $\mathbf{Q} : m, n \in \Re^2 \rightarrow \mathbf{q}(m, n) \in \Re^2$ which maps a scene point of coordinates m, n into the corresponding reflected point $\mathbf{q}(m, n)$ in the image plane. In the following,

we show that if it is possible to identify $N \geq 6$ points $\mathbf{p}_1$, $\mathbf{p}_2$, $\cdots$ $\mathbf{p}_N$ (as well as their corresponding image reflections $\mathbf{q}_1$, $\mathbf{q}_2$, $\cdots$ $\mathbf{q}_N$) in a neighborhood of $\mathbf{p}_o$, then the mapping $\mathbf{Q}$ can be estimated up to second-order and local shape around $\mathbf{r}_o$ can be recovered up to third-order.

In fact, a point $\mathbf{q}(m,n)$ can be developed in Taylor expansion around $\mathbf{q}_o$ as follows:

$$\begin{aligned}\mathbf{q}(m,n) \quad &\approx \quad \mathbf{q}_o + \dot{\mathbf{q}}_m m + \dot{\mathbf{q}}_n n + \\ &\qquad \frac{1}{2!}\ddot{\mathbf{q}}_{mm} m^2 + \frac{1}{2!}\ddot{\mathbf{q}}_{nn} n^2 + \ddot{\mathbf{q}}_{mn} mn, \end{aligned} \tag{3.117}$$

where the vectors $\dot{\mathbf{q}}_m$ and $\dot{\mathbf{q}}_n$ are the components of the Jacobian (call it $\mathbf{J}_Q$) of the mapping; the vectors $\ddot{\mathbf{q}}_{mm}$, $\ddot{\mathbf{q}}_{nn}$ and $\ddot{\mathbf{q}}_{mn}$ are the components of the Hessian (call it $\mathbf{H}_Q$) of the mapping. It can be shown that $\mathbf{J}_Q$ and $\mathbf{H}_Q$ can be estimated from Eq. (3.117) if the measurement of $N \geq 6$ points in a neighborhood of $\mathbf{p}_o$ (i.e., $\mathbf{p}_1$, $\mathbf{p}_2$, $\cdots$ $\mathbf{p}_N$) and their corresponding measurements ($\mathbf{q}_o$, $\mathbf{q}_1$, $\mathbf{q}_2$, $\cdots$ $\mathbf{q}_N$) are available. From $\mathbf{J}_Q$, the local shape of the mirror surface around $\mathbf{r}_o$ can be estimated up to second-order (by means of Eq. (3.98), Eq. (3.99), and Eq. (3.76)). From $\mathbf{H}_Q$, third-order surface parameters can be estimated (by means of Eq. (3.105)). $\mathbf{J}_Q$ and $\mathbf{H}_Q$ capture information equivalent to scale and orientation measurements around $\mathbf{q}_o$. Notice that the accuracy in estimating $\mathbf{J}_Q$ and $\mathbf{H}_Q$ depends on both the size of the neighborhood around $\mathbf{p}_o$ and the curvature of the mirror surface around $\mathbf{r}_o$. Finally, notice that if the second- and higher-order terms in expansion Eq. (3.117) can be neglected, $\mathbf{J}_Q$ can be estimated from just three points measured in the image plane. From $\mathbf{J}_Q$, local surface shape (up to second-order) around $\mathbf{r}_o$ can be estimated.

3.6 Experiments

We validated our theoretical results by recovering the local surface parameters of some real mirror objects. A Kodak DC290 digital camera with 1792×1200 pixel resolution was used to take the picture of a mirror surface reflecting a chosen ruled pattern. The mirror surface and camera were set at $30 - 50$ cm apart. The pattern was placed such that both pattern and its specular reflection were clearly visible from the camera (see Figure 3.10 or Figure 3.5). In order to test the reconstruction algorithms presented in and Table 3.6 respectively, we experimented with two types of patterns: one is the triangular pattern introduced in Section 3.5.3.2, the other, a checkerboard pattern as in Section 3.5.4.3. The camera and pattern were calibrated by taking advantage of the visible portion of the pattern and by using the Bouguet's calibration toolbox [9].

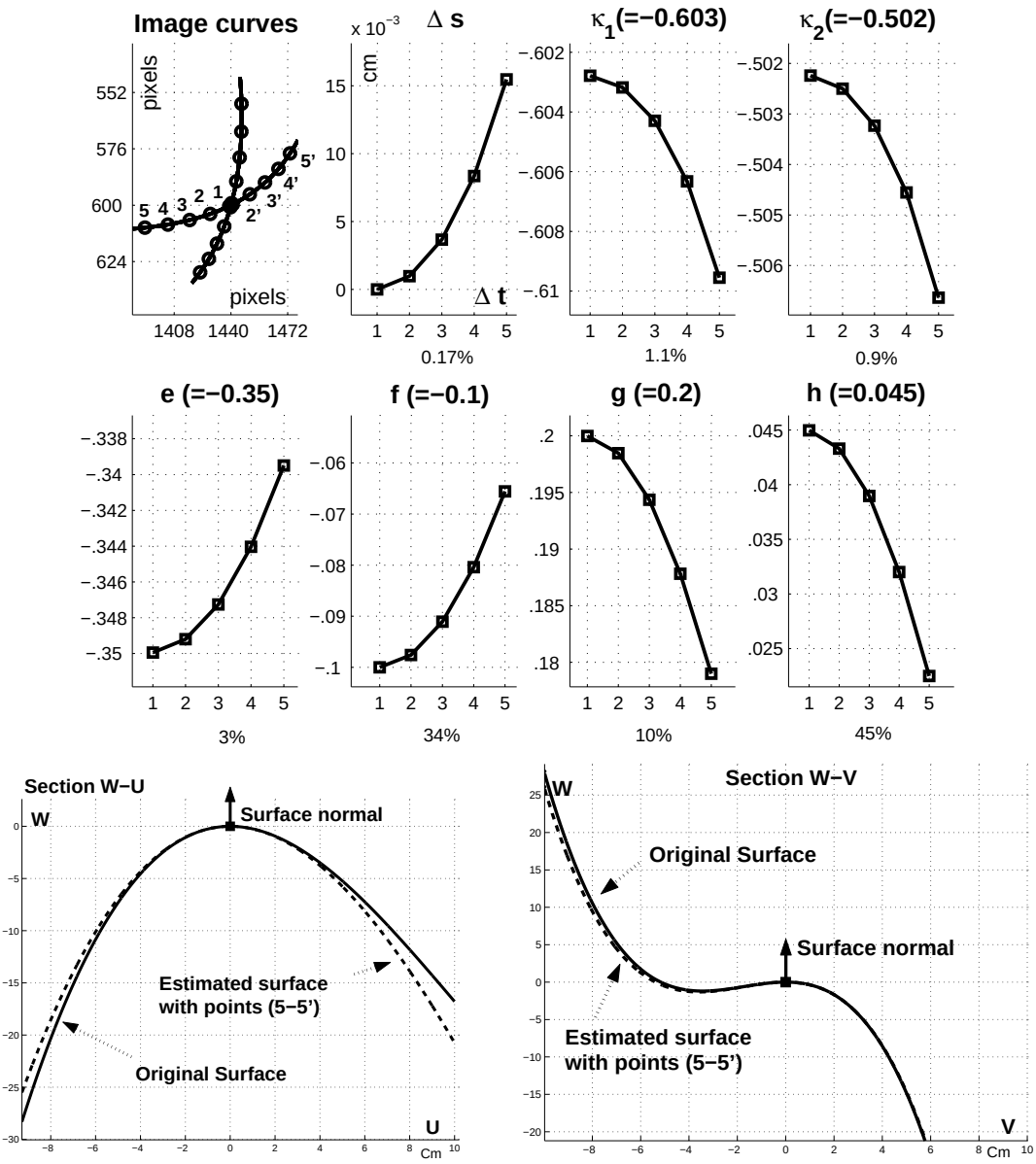


Figure 3.12: **Example of reconstruction error**

3.6.1 Results with Algorithm A1 - Measurement of Orientations

We validated algorithm A1 in Table 3.5 with four mirror surfaces: a plane (Fig. 3.13), a sphere (Fig. 3.14), a cylinder (Fig. 3.15), and a sauce pan lid (Fig. 3.16). We reconstructed $10 - 20$ surface points and normals for each of these mirror shapes. Correspondence between scene points and reflected points were established by hand. The curvature was recovered for the mirror sphere only. We tested the accuracy of the algorithm in estimating position and orientation of the mirror plane. We calculated the corresponding ground truth plane by attaching a calibrated pattern to the mirror plane and by recovering the position of 50 points on the pattern.

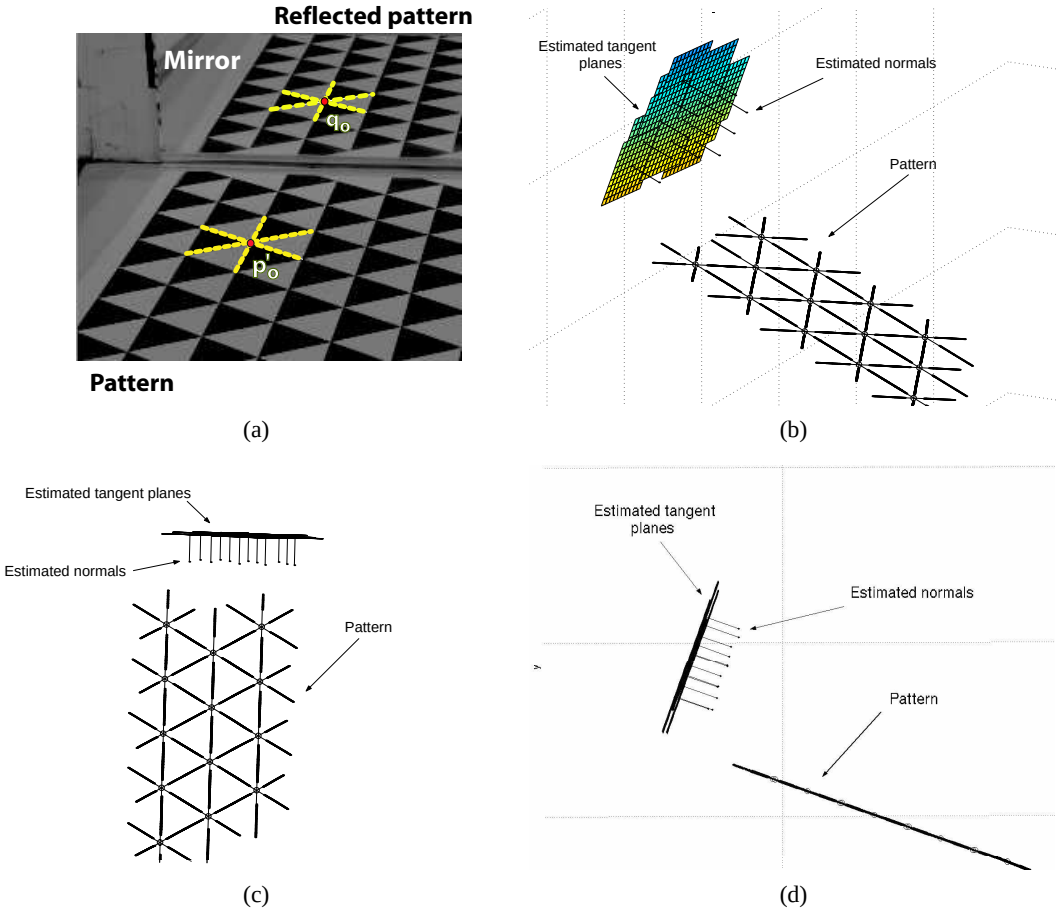
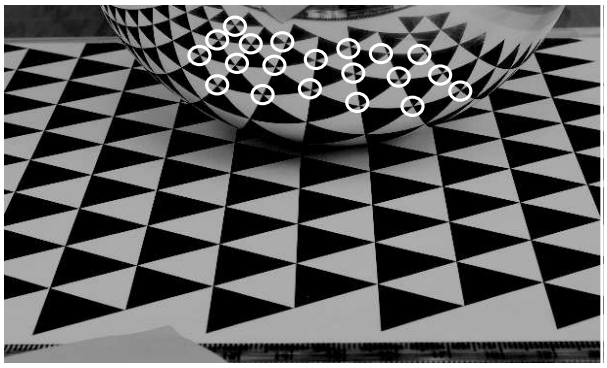
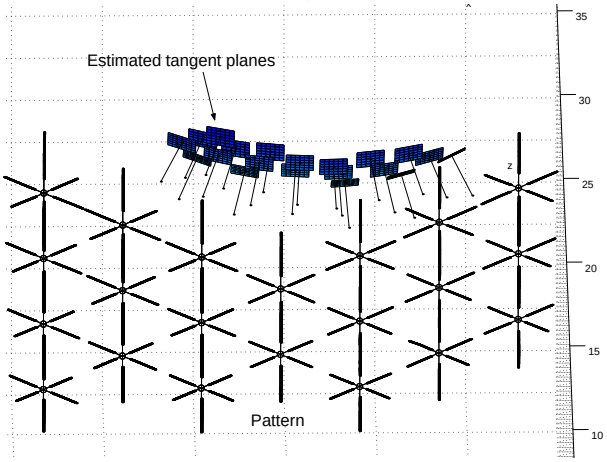


Figure 3.13: **Reconstruction of a planar mirror.** (a) A planar mirror placed orthogonally with respect to the ground plane. A triplet of pattern lines and the corresponding reflected triplet are highlighted with dashed lines. We reconstructed 12 surface points and normals on the mirror plane with algorithm A1 (see Table 3.5). The resulting mean position error (computed as average distance from the reconstructed points to the ground truth plane) was -0.048 cm with a standard deviation of 0.115 cm. The mean normal error (computed as the angle between ground truth plane normal and estimated normal) was 1.5×10^{-4}rad with a standard deviation of 6.5×10^{-4}rad. The reconstructed region was located at about 50 cm from the camera. (b) $3/4$ view of the reconstruction: we reconstructed 12 surface points and normals. For each reconstructed point, tangent planes are also plotted. (c) Top view. (d) Side view.

The ground truth plane was then obtained as the least-square plane fitting such set of points. The least square error was 0.062cm. Additionally, we tested the accuracy of the algorithm in estimating the curvature of the mirror sphere. The ground truth diameter of the sphere was measured with a caliper with accuracy 0.0013 cm. Table 3.7 shows the statistics of the reconstruction error attached to these two experiments (see rows $1-3$).



(a)



(b)

Figure 3.14: **Reconstruction of the sphere.** (a) A spherical mirror placed on the ground plane. We reconstructed 20 surface points and normals on the mirror sphere with algorithm A1 (see Table 3.5). The reconstructed points are highlighted with white circles. For each surface point we estimated the radius by means of Eq. (3.78). The ground truth radius was $r = 6.498$ cm ± 0.0013. The mean reconstructed radius was 6.83 cm and the standard deviation was 0.7 cm. The reconstructed region was located at a distance about 30 cm to the camera. (b) Top view of the reconstruction.

3.6.2 Results with Measurement of Orientations and Scale

We validated algorithm A2 in Table 3.6 with a specular teapot (see Figure 3.18), and a portion of car fender (see Figure 3.19). The recovery of the third-order surface surface parameters was validated in Figure 3.12 (bottom panel) using a synthetic mirror surface. We tested the accuracy of the algorithm in estimating the principal curvatures of the mirror cylinder (base of the teapot). The ground truth diameter of the cylinder was measured with a caliper with accuracy 0.0013 cm. Table 3.7 shows the statistics of the reconstruction error attached to this experiment (see rows $4 - 5$).

In Figure 3.17 we qualitatively compared the cylinder's reconstruction results obtained with the two

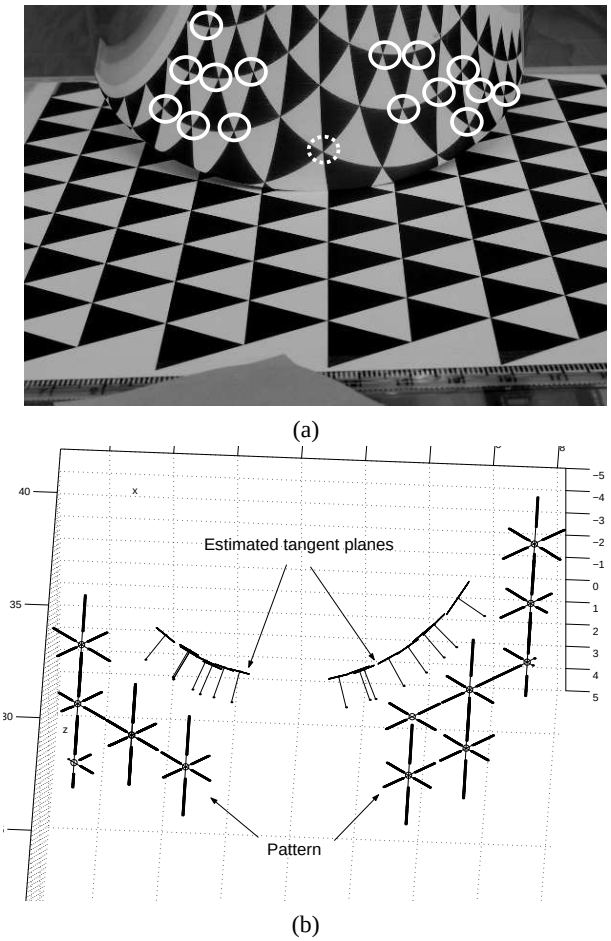


Figure 3.15: **Reconstruction of the cylinder.** (a) A cylinder placed with the main axis almost orthogonal to the ground plane. We reconstructed the surface (at the points highlighted with white circles) with algorithm A1. The dashed circle indicate an instance of point for which the reconstruction is highly inaccurate due to proximity to degenerate configuration (see Section 3.5.3.3 for details). (b) Top view of the reconstruction.

algorithms. Overall, the two methods exhibit similar performances in reconstructing position and surface normal at each intersecting point. However, the second approach is more advantageous than the three-line approach in that it can also estimate curvature parameters and requires a simpler scene structure.

3.7 Conclusions and Future Work

We analyzed the geometry underlying specular reflections of a known planar scene. We first presented the explicit relationship between a planar scene patch, the local shape of a mirror surface and its corresponding

Alg.	Surface	Fig.	Quantity	GT	Mean Error	Standard Dev.	N
A1	Plane	14	Point position	-	-0.048 cm	0.115 cm	12
A1	Plane	14	Normal orient.	-	1.5×10^{-4} rad	6.5×10^{-4} rad	12
A1	Sphere	15	Radius	6.49 cm	0.33 cm	0.7 cm	20
A2	Cylinder	18	κ_1	-0.152	-0.01	0.005	17
A2	Cylinder	18	κ_2	0	0.003	0.007	17

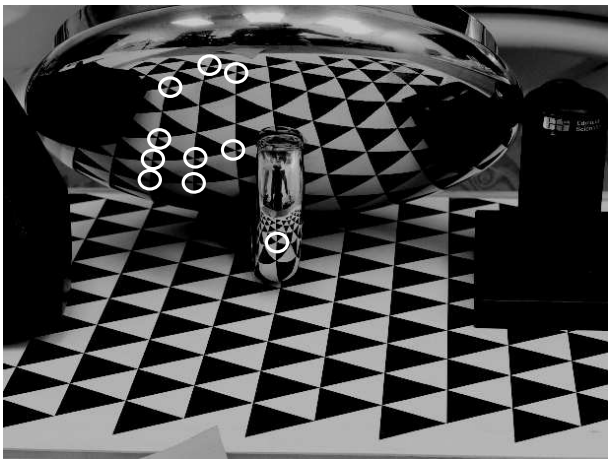
Table 3.7: Statistics of the reconstruction error for plane, sphere and cylinder. The first column shows which algorithm was used. The second one shows which surface was tested. The third one shows the figure illustrating the corresponding quantitative reconstruction results. The fourth one shows the geometrical quantity whose reconstruction accuracy was evaluated. The fifth, sixth and seventh ones show the corresponding ground truth value, the mean error and its standard deviation, respectively. The last one shows the number of points reconstructed.

specular reflection in the image. We studied several properties attached to this mapping and its degeneracies.

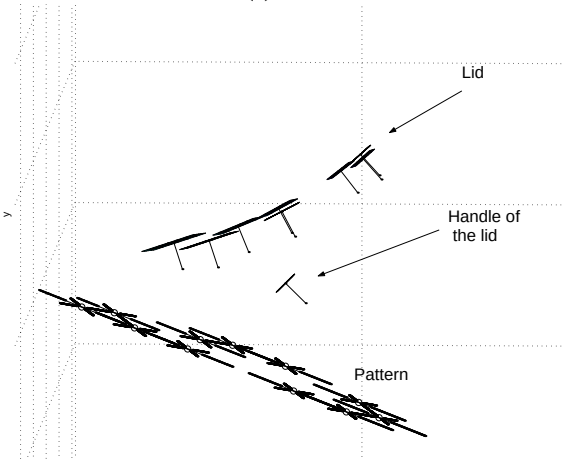
We used this analysis as a starting point for the reconstruction problem: torecover the shape of a specular surface from the reflections in one image. Under the assumption of an unknown mirror surface reflecting a known calibrated planar scene (e.g., a triangle-based pattern, a regular grid, or an arbitrary drawing) onto the image plane of a calibrated camera, we demonstrated that surface position and shape, up to third-order may be recovered. We started by assuming that the scene is composed of the simplest primary structure, namely lines intersecting at points. We assumed that measurements of orientation and curvature of each reflected curve are available. We proved that three orientation measurements are necessary and sufficient to recover the surface shape up to first-order (position and orientation). Second-order surface parameters may be recovered only up to an unknown parameter. Curvature measurements can be used as a further constraint. In a second step, we assumed that the scene is a grid of intersecting lines and that the measurements are both orientations and local scale of the reflected lines at the grid point. We proved that if orientation and local scale measurements along (at least) teo reflected curves are available, local surface shape can be fully recovered up to third-order accuracy. Finally, we generalized these results to the case of arbitrary planar scenes. We proved that local shape up to third-order can be extracted from the reflection of a planar scene patch observed in the image plane.

We validated our theoretical results with both numerical simulations and experiments with real surfaces and found that the method is practical and yields good quality reconstruction. We point out that since our reconstruction scheme allows local shape recovery around each reflected point, we obtain a “piece-wise” parabolic reconstruction. A robust estimation of the surface’s shape may ultimately be obtained by integrating such information.

These results can be considered as an ideal observer theory for local shape reconstruction from specular reflection and may be used as a complementary tool to investigate the abilities and limitations of the human



(a)



(b)

Figure 3.16: **Reconstruction of a sauce pan lid.** (a) A sauce pan lid placed with the handle touching the ground plane. We reconstructed the surface (at the points highlighted with white circles) with algorithm A1. Notice that one point belongs to the handle of the lid. (b) Side view of the reconstruction. Notice how the reconstructed point on the handle sticks out from the body of the lid.

visual system [57] [20].

Future work is needed to: i) calculate automatically the correspondence between pattern points and their reflected image points; and ii) calculate the shape from non-calibrated scenes. This effort will most likely require integrating additional cues, such as occluding boundaries, and some form of prior knowledge on the likely statistics of the scene geometry.

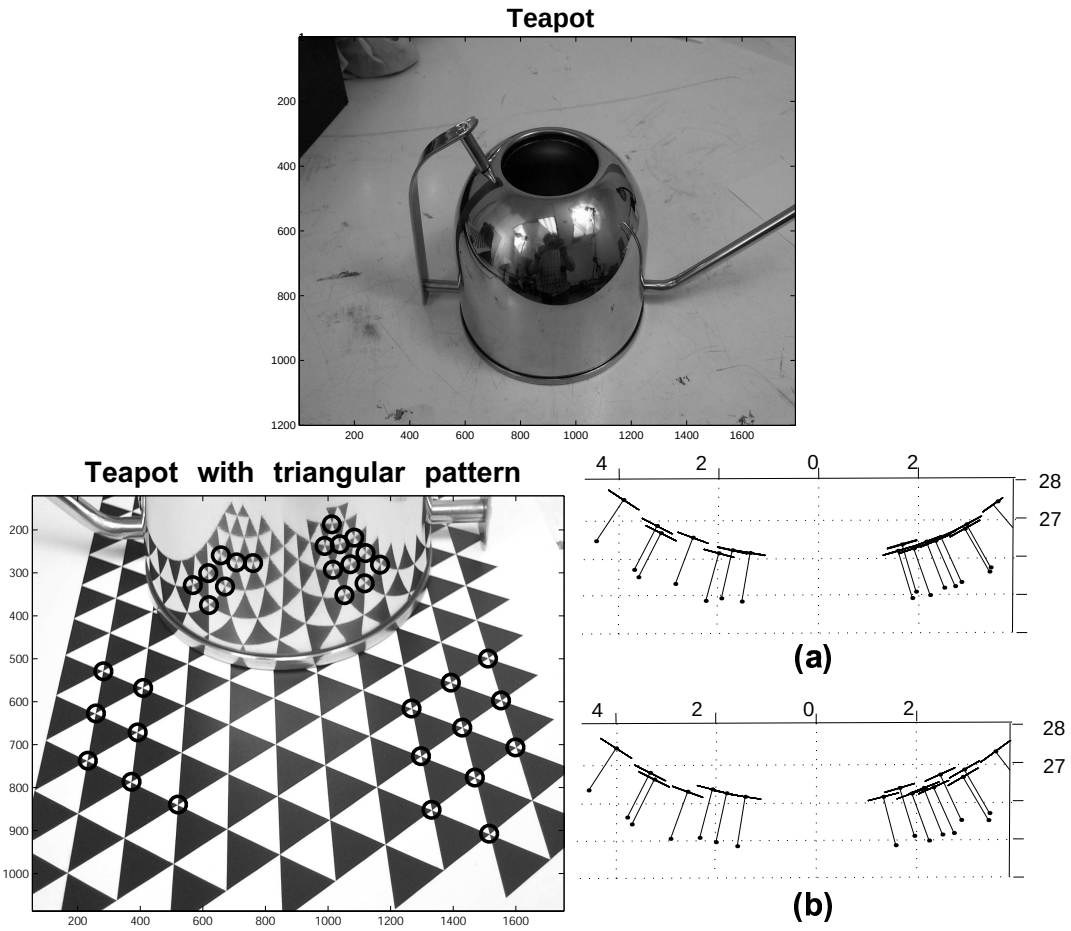


Figure 3.17: **Comparison of the two proposed reconstruction methods**. **Top panel:** a specular teapot. **Bottom left panel**: the two algorithms are tested at each point marked by circles. **Bottom right panel:** Reconstructed points and normal obtained by using three lines without scaling information obtained by mark points (a) and two lines with scaling information obtained by mark points (b). Notice that the performances of the two reconstructed points and normals are comparable. By means of algorithm A2, we recovered the principal curvatures of the cylinder. The diameter ground truth was 13.158 cm ± 0.0013. The estimated average principal curvatures were $\kappa_1 = -0.153 \pm 0.005$ and $\kappa_2 = 0.003 \pm 0.007$, which correspond to an average estimated cylinder diameter of 13.097 cm.

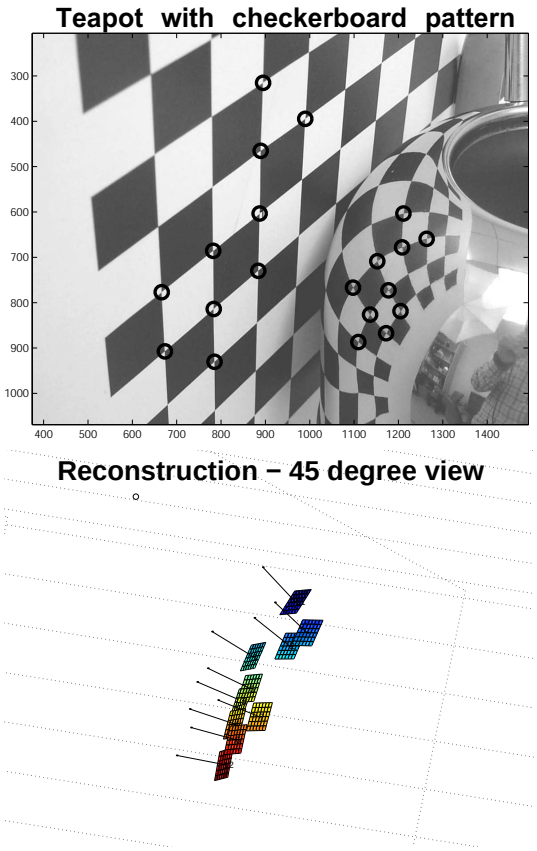


Figure 3.18: **Teapot Experiment. Top panel:** top part of a teapot. We reconstructed the surface at the points highlighted with black circles. **Bottom panel:** 45^o view of the reconstruction.

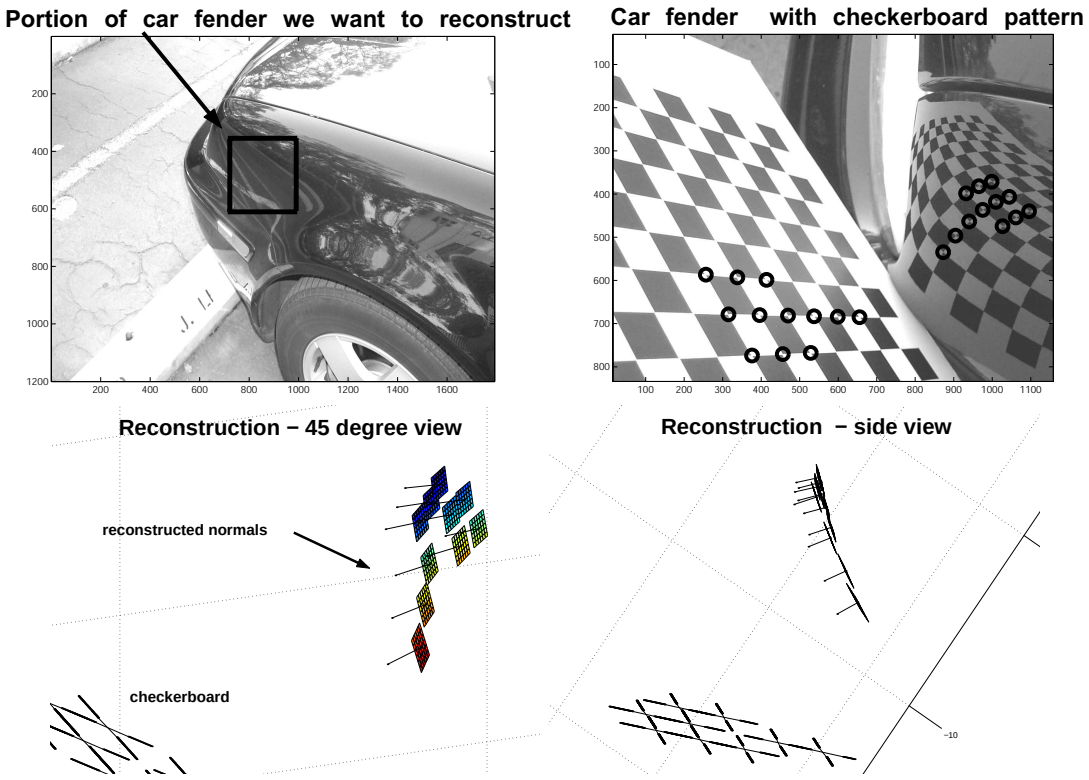


Figure 3.19: **Car Experiment. Top Left panel:** top portion of car fender we wanted to inspect. **Top Right panel:** we reconstructed the surface at the points highlighted with black circles. Notice that one image suffices to calibrate both camera and scene and recover the mirror's shape at the marked points. **Bottom left panel:** 45^o view of the reconstruction. **Bottom right panel:** side view of the reconstruction.

Chapter 4

Human Perception of Specular surfaces

4.1 Introduction

A sense of three dimensional shape may be perceived by looking at a two dimensional image of an object. This perception arises due to different cues such as contour, shading, perspective and texture. When viewing a picture of a specular object such as a silver vase, one additional piece of information is represented by the reflection of the environment. A deformed picture of the environment is seen in the surface of the object, and the amount and type of deformation depend on its shape. Can specularities be used as a visual cue for shape perception, especially in the absence of other cues?

There has been extensive research in the past several years to understand the geometrical relationship between shape and specular reflections, and many computational models have been proposed. Previous contributions are summarized in Section 3.1.2. Our own contribution is presented in Chapter 3. Our aim was to recover the local shape of an unknown smooth specular surface through the reflection of a known scene patch. In our approach, a surface reflects a scene patch onto the image plane of a calibrated camera. For instance, the scene can be a grid of intersecting lines. Thus, a mapping from the scene grid to the reflected grid in the image is defined. Such mapping changes the *orientation* of the grid lines and *stretches* the grid step, modifying the local scale of the pattern. We proved that the local shape of the surface can be recovered if and only if: i) the scene is calibrated; and ii) at least two orientations and local scale measurement (or equivalent information) are available at the same point in the image. Our results can be considered as an *ideal observer theory* for local shape reconstruction from specular reflection.

What happens when the surrounding world is unknown? The ideal observer analysis tells us that the problem is locally under-constrained and many solutions are possible. Thus, it may be useful to study the problem psychophysically and answer questions of the sort: how do humans cope with the problem of perception of the shape of specular surfaces, given that most of time, we do not have the exact geometrical information

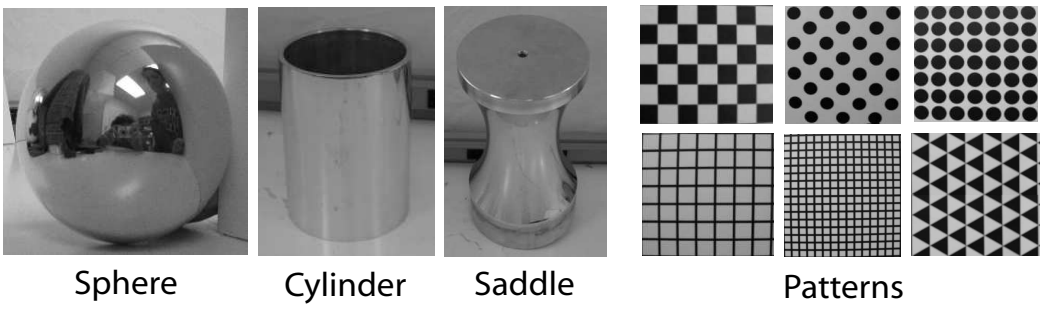


Figure 4.1: **Left:** the three shapes used in the experiment. **Right:** the six patterns used in the experiment

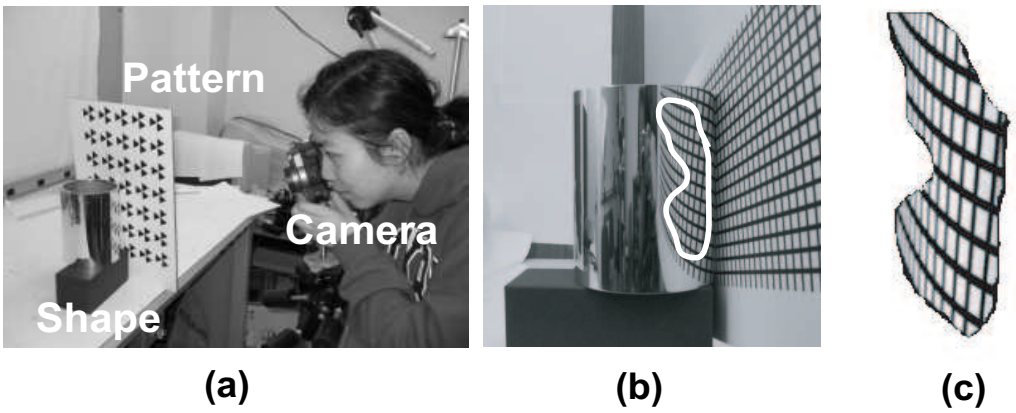


Figure 4.2: **(a)**. The setup: a camera takes a picture of a mirror shape reflecting a pattern. **(b)** Picture of the mirror shape reflecting the pattern: a patch of the reflection of the pattern is cropped from the picture. **(c)** The cropped patch does not include any visual cues besides a portion of the reflected pattern.

about the world? How useful are specular reflections when other visual cues are absent?

Shape perception from visual cues has been largely discussed in the vision literature. Models of visual processes, however, have tended to regard specular reflections as noise disturbing the underlying object geometry. The fact that highlights carry useful information was first discussed by Beck [1] and Koenderink [36]. In the early nineties, Blake and Bulthoff [5] showed that the human visual system estimates shape and quality of a shiny object when the highlights are viewed stereoscopically. Works by Todd [73], Lu [44] and Nayar [48] are along this line of thought. Dror, Adelson, et al. [22] explored whether statistics of real world illumination could be helpful in recognizing optical reflectance properties of shiny materials. Surprisingly, the reflection of a surrounding scene as a cue for shape perception has hardly been explored psychophysically. Exceptions are recent works by Fleming [20, 21]. Through the present study we would like to give some insight into this problem and address the following questions: i) to what extent can the human visual system use specular reflections in the absence of other visual cues; and ii) what are the underlying computational strategies for

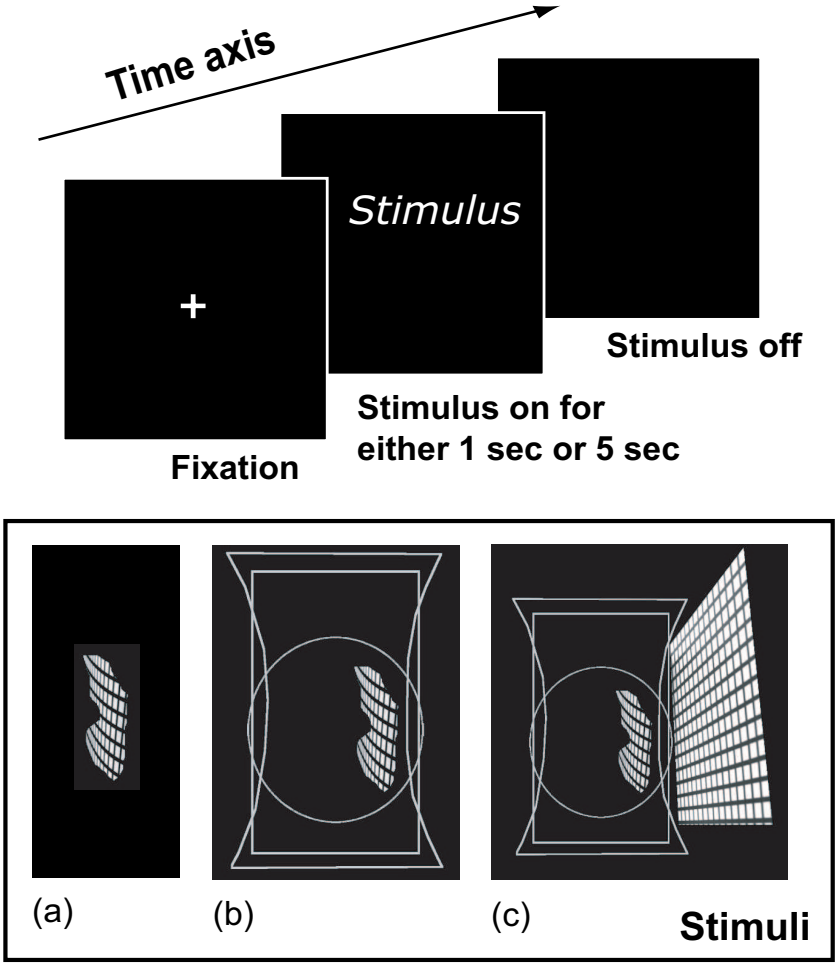


Figure 4.3: **Top:** Each trial of the experiment is carried out as follows: A fixation cross is presented at the beginning of the trial for 240 msec. A stimulus (see **Bottom** panel) is then presented for either 1 sec or 5 sec. Subjects are instructed to respond to the shape of the mirror as fast and as accurately as possible by pressing one of the three designated keys. **Bottom (a,b,c)** Examples of stimuli presented during experiment 1,2 and 3, respectively.

this task?

4.1.1 Chapter Organization

The rest of the chapter is organized as follows. In Section 4.2, we describe the experimental setup and method. In Section 4.3 we present our results. In Section 4.4, we analyze and discuss the results. We conclude the chapter with final remarks in Section 4.5.

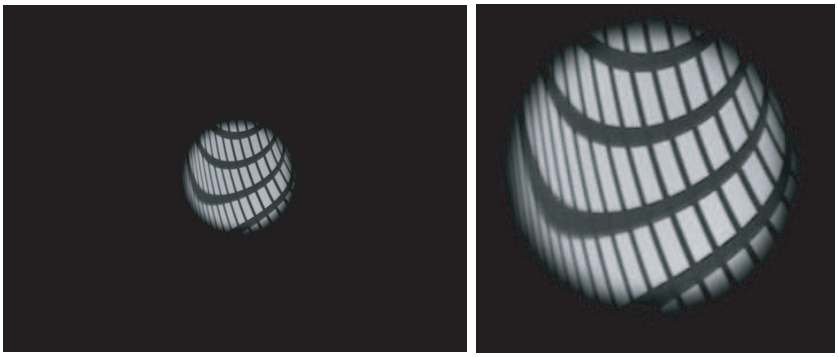


Figure 4.4: **Left:** Example of stimulus used for the control experiment. **Right:** Detail of the stimulus.

4.2 Methods

Our goal is to study whether the human visual system uses specular reflection as a cue for shape perception. We asked ten naive human subjects to discriminate images of mirror surfaces of qualitatively different shapes: a sphere, a cylinder and a saddle (i.e., with positive, zero, and negative Gaussian, respectively (Fig. 4.1, left). Such shapes reflected the same scene with distinctly different distortions.

There were three consecutive experiments in our study and one control experiment. In the first experiment, the stimuli consisted of 144 photographs of large patches of each mirror surface (Fig. 4.2, (a)). Each patch contained the reflection of one of the six regular patterns (Fig. 4.1, right). The patterns were shown to the subjects in the familiarization phase before the actual experiment. Each patch was obtained by vignetting one of the photographs. We used irregularly shaped boundaries in order to eliminate occluding boundary information (Fig. 4.2, (b) and (c)). Each patch comprehended $25 - 35\%$ of the overall observable surface area. This guaranteed that the geometrical information of the surface shape provided by each stimulus was sufficiently rich. An example of a stimulus is in Fig. 4.4, (a). Each stimulus was viewed monocularly and centrally in a standard computer monitor and subtends an average visual angle of 20^o. Each trial was made up of three main steps as explained in Fig. 4.4, top. We presented each stimulus with two different presentation times ($1sec$ or $5sec$).

In the second experiment, we used the same set of stimuli as in Experiment 1, except for an additional piece of information: three overlapped silhouettes of the shapes were presented together with the cropped patch (Fig. 4.4, (b)). Thus, subjects were provided the approximate position of the patch within the shapes. In the third experiment, we used the same set of stimuli as in Experiment 2. Again, an additional piece of information was given: the position of the pattern producing the reflection was shown (together with the cropped patch and three overlapped silhouettes, Fig. 4.4, (c)). Hence, subjects were presented with the full geometrical structure, with the only unknown being the actual shape which is yielding the patch.

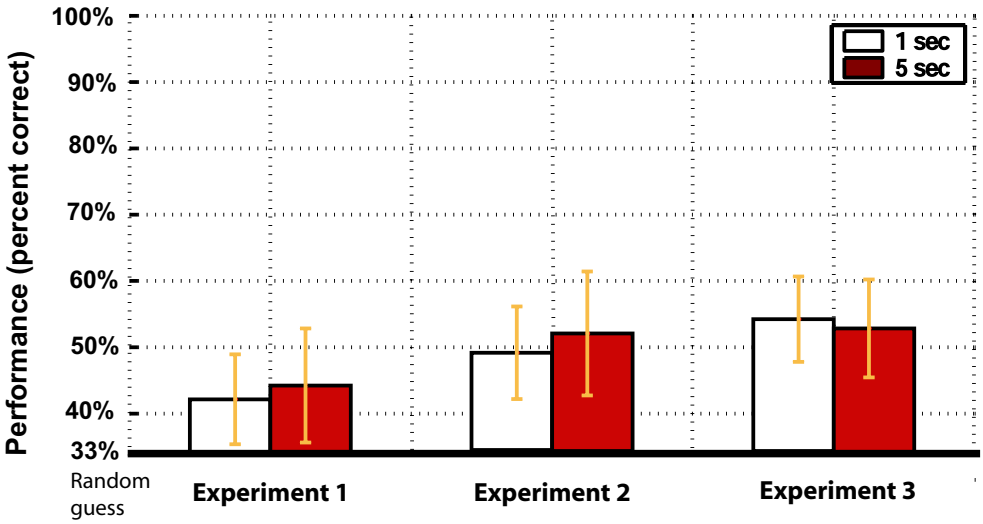


Figure 4.5: Performance summary of Experiment 1, 2 and 3. 33% means chance level performance. Note the average performance of all ten subjects are only slightly better than chance.

We ran the three experiments in a consecutive order for each subject. This was done to rule out the possibility that subjects may have had memory and taken advantage of additional information provided by Experiments 2 and 3. During each experiment, all 288 stimuli (144 images $\times 2$ different presentation times) were presented in random order, counter-balanced across all ten subjects. A familiarization phase preceded the experiments, during which subjects were shown the six patterns. In addition, subjects were given a short period to practice keyboard responses for the three different mirror shapes (in the absence of specular reflection).

In a separate control experiment we tested subjects' performance (as from the first experiment) with patches having circular boundaries rather than irregularly shaped boundaries. The circular boundary was obtained by applying a radial fading effect ranging from fully transparent (inside the circle) to solid black (outside the circle). The goal was to give the subjects the perception of a black panel located in front of the object, with a hole in the middle, through which the object's specular reflections are visible. See Fig. 4.4.

4.3 Results

The results of the three experiments are summarized in Fig. 4.5. Performance (in percent correct) is shown for Experiments 1, 2 and 3. This result indicates that: i) our subjects are only slightly better than chance in discriminating the three mirror shapes; ii) performance that corresponded to presentation time of one second and five seconds is not statistically different (t-test: p value > 0.05). This suggests the processing of such

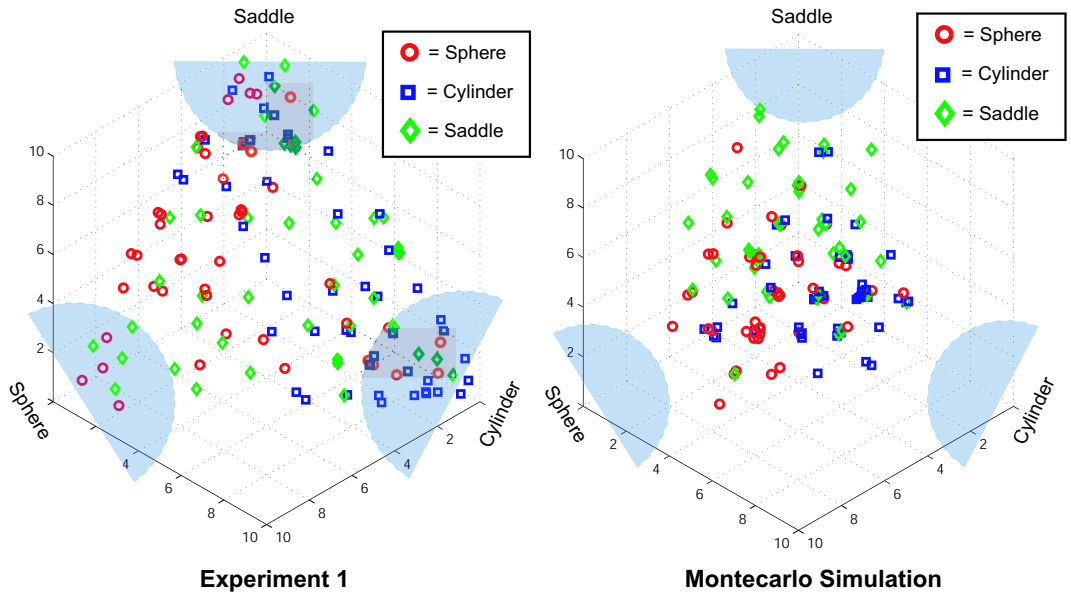


Figure 4.6: **Left.** Distribution of 144 stimuli for Experiment 1 and presentation time 1sec. Each point corresponds to one of the 144 stimuli. Its position depends on the response patterns generated by ten subjects. First, second and third coordinates of the point are the number of subjects who consider the stimulus as the specular reflection yielded by a sphere, cylinder and saddle respectively. For instance, a stimulus chosen by 10 subjects as the specular reflection yielded by a sphere, corresponds to a point with coordinates $(10, 0, 0)$. Regions around the 3 vertices $(10, 0, 0)$, $(0, 10, 0)$ and $(0, 0, 10)$ contain stimuli that exhibit high agreement among subjects. Those regions are in blue. The color code indicates the *actual* shape of the surface yielding that particular stimulus. **Right.** Monte-Carlo simulation to the 144 stimuli showing that if different subjects' responses were uncorrelated, we would observe a completely different distribution.

tasks is fast; and iii) performance slightly improves from Experiment 1 to Experiment 2 (t-test: p value < 0.05). Thus, the knowledge of the location of the patch within the shape silhouettes seems to be helpful. Performance is, however, not significantly different between Experiment 2 and Experiment 3, suggesting that a richer geometrical structure has little effect in the way shape is recognized from reflection. This observation also suggests that the improvement from Experiment 1 to 2 is not due to practice.

To rule out the possibility that subjects might treat irregularly shaped outlines as occluding boundaries (thus, as a intrinsic properties of the object), we conducted the control experiment described in Sec. 4.2. In this experiment, the circular contour gives the subject the illusion that the reflection is observed behind a black panel through a circular hole. We found that there is no substantial difference of performances between this control experiment and Experiment 1: the mean percentage correctness was 0.42 with a standard deviation of 0.05.

While subjects are almost at chance in discriminating shape differences from reflection, we find that responses are highly correlated. In other words, subjects tend to make decisions in a very consistent manner. We may visualize this observation through Fig. 4.6 (details are in the caption of Fig. 4.6). Let us call *high*

consistency regions the volumes of radius $R = 10/3 = 3.3$, centered at each of the vertices $(10, 0, 0)$, $(0, 10, 0)$ and $(0, 0, 10)$ of the diagram. By inspection of Fig. 4.6, it is clear that stimuli tend to cluster toward the three vertices, revealing that there is high agreement amongst subjects' answers. We can contrast this with an uncorrelated Monte Carlo simulation of the responses. We take the observed average performance of Experiment 1 (44%), and generate a simulated response pattern similar to the one in Fig. 4.6, Left panel. On average, only 0.05% of the 144 stimuli fall into the three *high consistency regions*, indicated by the light blue semi-spheres (Fig. 4.6, Right panel). To the contrary, 29% of our actual stimuli are contained in the high consistency regions, revealing strong agreement in subjects' answers, both correctly and incorrectly. We obtain similar results for Experiments 2 and 3. Fig. 4.7 shows examples of stimuli contained in the three consistency regions as well as in the regions in between.

4.4 Analysis

The conclusions drawn in Sec. 4.3 lead us to consider the nature of the mechanism that may underlie this perception. If we consider the stimuli which give consistent shape perception (Fig. 4.7, first three rows) we see patterns of deformation that are similar within each category and different across categories. Either one or two patterns seem to account for the perception of each shape (see Fig. 4.8). Stimuli for which there is no consistent perception (Fig. 4.7, last row) may be associated to combinations of the 'pure' patterns of Fig. 4.8. One possible interpretation of this finding is that our subjects, rather than 'computing' a percept from each image based on geometrical considerations, may be associating a shape to each pattern in a stereotypical way, akin to pattern-matching. This behavior is reasonable since, as suggested by our analysis [55], the information that is available from our stimuli is ambiguous.

If shape perception is the result of pattern-matching, it would be reasonable to believe that the 'fundamental' patterns of specular surfaces have been learned from experiences, and correspond to highly likely scene-surface configurations. A mirror cylinder standing on a horizontal textured plane would be one such example. However, we find that many of the stimuli for which there is a consistent and wrong perception correspond to such situations. Therefore, we consider this an unlikely explanation. Another possibility is that our subjects are analyzing our stimuli as if they were images of curved textured surfaces rather than of curved mirrors. Notice that in Fig. 4.4, (c) and Fig. 4.7, our stimuli often do not appear to be mirror-like. A qualitative analysis of the patterns in Fig. 4.8 is consistent with this interpretation, if one assumes a textured surface where the principal lines of the texture are roughly aligned with the principal curvatures of the surface. This leads to the intriguing hypothesis that shape perception from specularities and texture deformations [81, 72, 34, 32, 46, 3] seem to share a common basis.

If the texture-based pattern-matching is a strategy that the visual system uses to perceive shape from reflection, what would this tell us about the underlying computation pathways that are used to perform such tasks? Moreover, can we infer from this that shape perception from texture and from mirrors share some common computation or strategy? How is such sharing of resources mapped out? Where do they merge and where do they branch off? All these questions are highly important and crucial to further our understanding of the visual system. In addition, it might shed more light on how to engineer more powerful and efficient models for shape perception.

A natural question is whether such performance would be different if one uses reflections of real world scenes rather than regular synthetic patterns. The reason is twofold. First, our patterns were simple geometrical forms. This could have led subjects to misinterpret our stimuli as textures rather than actual specular reflections. Second, the human visual system may exploit prior knowledge on regularities of real world structure (i.e., statistics of real world images). In fact, by identifying in specular reflections some degree of deviation from such statistics, the human visual system may become capable of successfully interpreting the shape of specular surfaces. We would like to address these questions in the future.

We conclude this section by comparing our results with those in [20]. Fleming and colleagues found that the shape of mirror surfaces is readily perceived, even when the only cue is specular reflections. They argue that specular reflections exhibit a different pattern of compression than surface texture. This feature would allow the human visual system to discriminate between these two cases. Hence, the pattern of compression would represent a cue for shape from specularities. We interpret this apparent opposite result by pointing out that a number of different cues might be integrated by the human visual system in their experiment. For instance, specular reflections show different behaviors depending on whether they appear on smoothly curved regions, or in proximity of occluding contours, or in regions of high curvature. In each case, specular reflection would provide the visual system with different information about the geometry of the surface. This is particularly true when the specular object has complex topology. Therefore, these indirect shape cues might also have contributed in the perception of shape in their experiments. Clearly, much additional work is needed to reconcile these findings to understand the contributions of these different factors, and in turn the underlying computational strategies.

4.5 Conclusions

We have studied how humans perceive shape from mirror reflections. We use six regular synthetic patterns and their mirror reflections in three different shapes: sphere, cylinder and saddle. Our results indicate that mirror reflection is only a very weak cue to shape perception in the absence of other visual cues. A closer

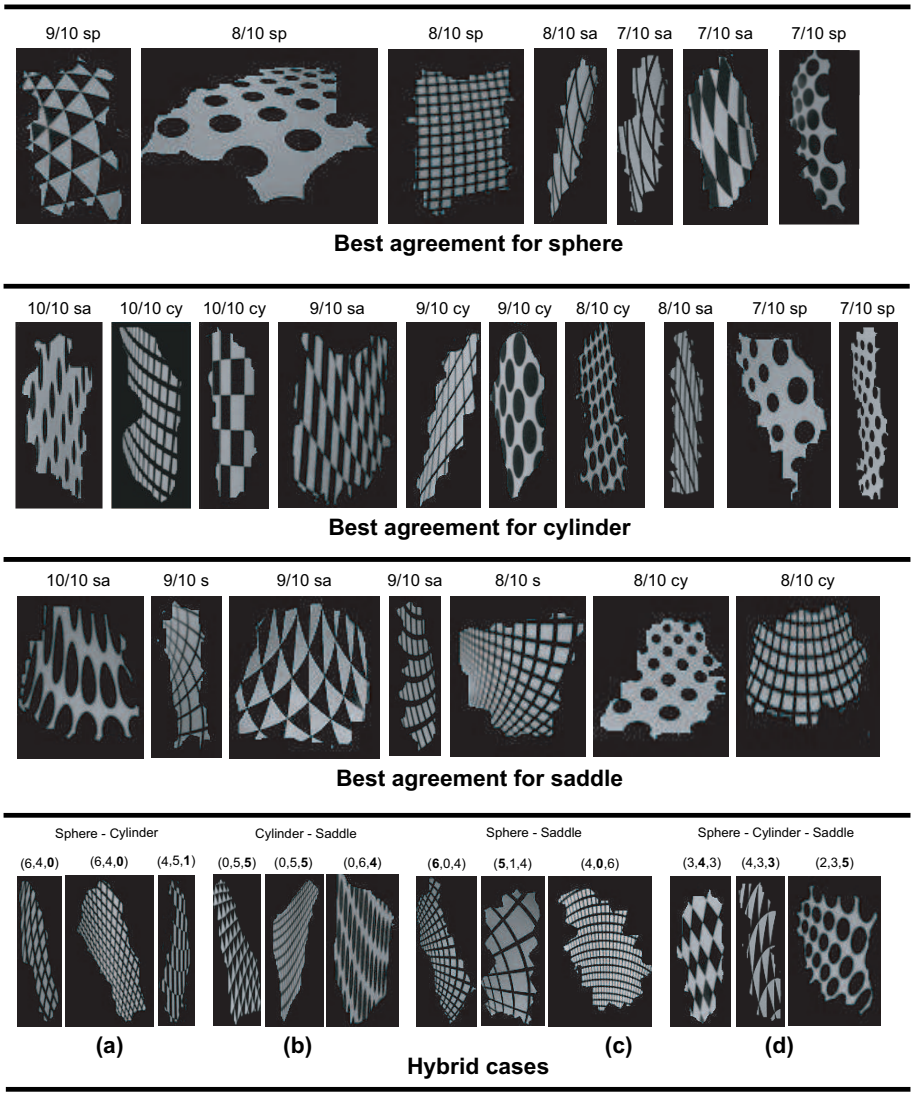


Figure 4.7: **First panel (from the top):** Examples of stimuli most frequently perceived as from the sphere shape (i.e., examples from the *high consistency region* of sphere in Fig. 4.6). The patches are sorted according to the score. The text above each stimulus (e.g., $8/10$ sa) indicates the score fraction (e.g., eight subjects out of ten agreed responded as a sphere) followed by the label of the actual shape of the surface attached to that patch (e.g., saddle). **Second, Third:** Examples of stimuli most frequently perceived as from the cylinder and saddle respectively. **Fourth-(a):** Example of stimuli contained in the region between sphere and cylinder (see also Fig. 4.6). These stimuli are consistently classified as either sphere or cylinder. Each stimulus is labelled by a vector as described in the caption of Fig. 4.6. E.g., $(6, 4, \mathbf{0})$ means that this stimulus is chosen as from sphere by 6 subjects, cylinder by 4, and saddle by none. Bold font of 0 indicates the actual shape of which the stimulus is reflected, in this case, a saddle. The order of shape for the responses is sphere, cylinder, saddle. **Fourth-(b,c,d):** Examples of stimuli (see also Fig. 4.6) contained in the region between cylinder and saddle, sphere and saddle, and central region, respectively, the latter having obtained the lowest agreement among subjects.

look at the response pattern of our ten subjects reveals that there might be a set of underlying *templates* that subjects use for shape perception from reflections. These templates follow closely the deformation patterns

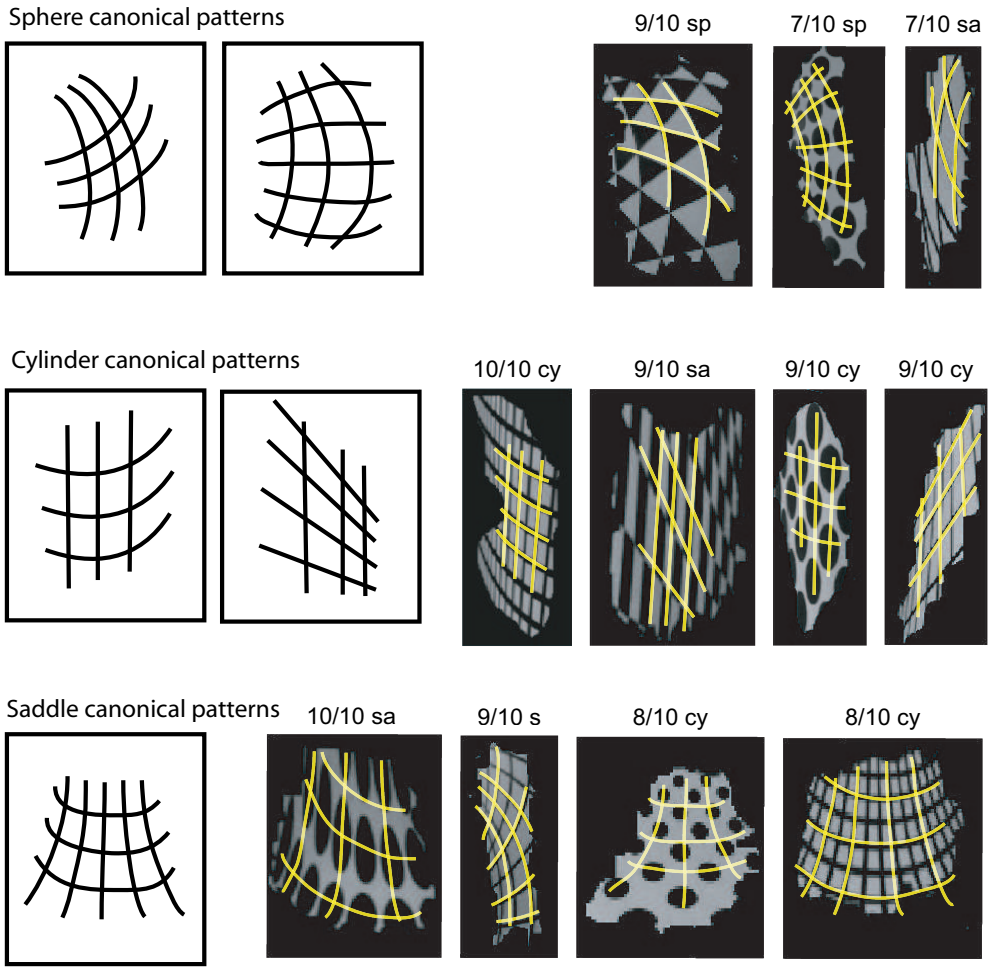


Figure 4.8: **Canonical patterns. Top:** Stimuli that are most frequently perceived as from sphere shape reflection seem to exhibit features that can be connected in either of the two template patterns on the left side of the panel. **Middle and Bottom:** Similar conclusion can be drawn for stimuli that are most frequently perceived as from cylinder and saddle shapes, respectively.

of texture on shapes, rather than reflection in shapes. Future work may include establishing a more complete phenomenology of human perception of specular surfaces addressing the questions: to what extent do humans perceive qualitative and metric shape differences? Do they make use of any knowledge (metric, statistical) about ambient structure? Are some ambient structures more informative then others? Do additional cues (such as contours, texture, shading) help in ways that are not purely summative?

Chapter 5

Conclusions

In this thesis, we addressed the issue of measuring automatically the shape of physical objects for 3D modeling. Our work was motivated by the goal of developing algorithms that: i) are inexpensive, fast and accurate; ii) can handle objects with arbitrary appearance properties and shape; and iii) need little or no user intervention.

In the first part, we presented a novel 3D reconstruction technique which makes use of minimal and inexpensive equipment. The technique is based on the idea of refining a conservative estimate of the shape of an object. This is done by observing the shadows on the object when it is lit by a known point light source. We proved that a well defined portion of volume can be removed in a conservative manner from the current object estimate. We showed that this insight leads to an algorithm that can work correctly on real images. We called this method *shadow carving*.

Our main assumption is that one can estimate shadows conservatively - i.e., a shadow may not be detected but whatever is labelled as shadow is indeed a shadow. No assumptions about the object topology are necessary. Tangent plane discontinuities over the objects surface have been supposed to be detectable. We showed that shadow carving improves previous work on shape from shadow in that it is more robust with respect to the classification of shadow regions and is not restricted to 2.5D terrain surfaces, but rather it may be applied to measuring the objects in the round.

In order to validate our theory, we implemented a reconstruction system that combines information from silhouettes and shadows. The new system uses inexpensive digital cameras and lamps. Our experiments with real and synthetic objects confirmed that the property of conservative carving is achievable in practice and showed that shadow carving produces a much better surface estimate that shape from silhouettes alone. This improved estimate is suitable for further refinement by any shape estimation method that works well in local regions of the surface.

It is well known that 3D scanning systems handle highly reflective surfaces only poorly. Specular reflec-

tions are often considered a nuisance for common 3D reconstruction techniques. In the second part of this thesis, we showed that it is possible to promote mirror reflections from ”noise” to ”signal”.

We analyzed the geometry underlying specular reflections of a known planar scene. We first presented the differential relationship between a planar scene patch, the local shape of a mirror surface and its corresponding specular reflection in the image. We studied several properties attached to this mapping and its degeneracies.

We used this analysis as a starting point for the reconstruction problem: recover the shape of a specular surface from the reflections in one image. Under the assumption of an unknown mirror surface reflecting a known calibrated planar scene (e.g. a triangle-based pattern, a regular grid or an arbitrary drawing) onto the image plane of a calibrated camera, we demonstrated that surface position and shape up to third order may be recovered. We started by assuming that the scene is composed of the simplest primary structure, namely lines intersecting at points. We assumed that measurements of orientation and curvature of each reflected curve are available. We proved that 3 orientation measurements are necessary and sufficient to recover the surface shape up to first order (position and orientation). Second order surface parameters may be recovered only up to an unknown parameter. Curvature measurements may be used as a further constraint. In a second step, we assumed that the scene is a grid of intersecting lines and that the measurements are both orientations and local scale of the reflected lines at the grid point. We proved that if orientation and local scale measurements along (at least) 2 reflected curves are available, local surface shape can be fully recovered up to third order accuracy. Finally, we generalized these results to the case of arbitrary planar scenes. We proved that local shape up to third order can be extracted from the reflection of a planar scene patch observed in the image plane.

We validated our theoretical results with both numerical simulations and experiments with real surfaces and found that the method is practical and yields good quality reconstruction. Since our reconstruction scheme allows local shape recovery around each reflected point, we obtain a “piece-wise” parabolic reconstruction. A robust estimation of the surface’s shape may be ultimately obtained by integrating such information.

Future work is needed to i) calculate automatically the correspondence between pattern points and their reflected image points; and, ii) calculate the shape from un-calibrated scenes. This effort will most likely require integrating additional cues, such as occluding boundaries, and some form of prior knowledge on the likely statistics of the scene geometry.

An additional goal of this thesis was advancing our understanding of human perception of shape from reflections. Surprisingly, little is known to what extent highlights and specular reflections carry useful infor-

mation for shape perception. We used psychophysics to study this capability. We used six regular synthetic patterns and their mirror reflections in three different shapes: sphere, cylinder and saddle. Our results indicated that mirror reflection is only a very weak cue to shape perception in the absence of other visual cues. Closer look at the response pattern of our ten subjects revealed that there might be a set of underlying *templates* that subjects use for shape perception from reflections. These templates follow closely the deformation patterns of texture on shapes, rather than reflection in shapes. Future work may include further investigation of the scope of this hypothesis using natural scene reflections.

Bibliography

[1] J. Beck and S. Prazdny. Highlights and the perception of glossiness. In *Perception and Psychoph.*, volume 30, pages 407–410, 1981.

[2] R. Bhotika, D. Fleet, and K. Kutulakos. A probabilistic theory of occupancy and emptiness. *Proc. of the European Conference on Computer Vision*, pages 112–132, 2002.

[3] A. Blake. Specular stereo. In *IJCAI*, pages 973–976, 1985.

[4] A. Blake and G. Brelstaff. Geometry from specularities. In *Proc. of Int. Conf. of Computer Vision*, pages 394–403, 1988.

[5] A. Blake and H. Bülthoff. Does the brain know the physics of specular reflection? In *Nature*, volume 343, pages 165–168, 1990.

[6] J. De Bonet and P. Viola. Roxels: Responsibility weighted 3D volume reconstruction. *Proc. of the International Conference on Computer Vision*, pages 418–425, 1999.

[7] T. Bonfort and P. Sturm. Voxel carving for specular surfaces. In *Proc. of Int. Conf. of Computer Vision*, pages 394–403, 2003.

[8] M. Born and E. Wolf. *Principle of optics – Electromagnetic theory and propagation*. OPergamon Press, 1965.

[9] J. Bouguet. *http://www.vision.caltech.edu/bouguetj/*.

[10] E. Boyer and M. Berger. 3D surface reconstruction using occluding contours. *International Journal of Computer Vision*, 22(3):219–233, 1997.

[11] A. Broadhurst and R. Cipolla. A probabilistic framework for space carving. *Proc. of the International Conference on Computer Vision*, pages 388–393, 2001.

[12] M. D. Carmo. *Differential geometry of curves and surfaces*. Prentice-Hall, 1976.

[13] M. Chen and J. Arvo. Theory and application of specular path perturbation. *ACM Transactions on Graphics*, 19:246–278, 2000.

[14] R. Cipolla and A. Blake. Surface shape from the deformation of apparent contours. *International Journal of Computer Vision*, 9(2):99–111, 1992.

[15] R. Cipolla and P. Giblin. *Visual motion of curves and surfaces*. Cambridge University Press, 2000.

[16] M. Daum and G. Dudek. On 3-D surface reconstruction using shape from shadows. In *Proc. of Computer Society Conference on Computer Vision and Pattern Recognition*, pages 461–468, Santa Barbara, CA, June 1998.

[17] P. Eisert, E. Steinbach, and B. Girod. Multi-hypothesis, volumetric reconstruction of 3D objects from multiple calibrated camera views. *Proc. of the International Conference on Acoustics, Speech, and Signal Processing*, pages 3509–3512, 1999.

[18] M. Farouk, I. El-Rifai, S. El-Tayar, H. El-Shishiny, M. Hosny, M. El-Rayes, J. Gomes, F. Giordano, H. Rushmeier, F. Bernardini, and K. Magerlein. Scanning and processing 3D objects for web display. In *Proc. of the International Conference on 3DIM*, pages 310–317, 2003.

[19] A. Fitzgibbon, G. Cross, and A. Zisserman. Automatic 3D model construction for turn-table sequences. In *Proceedings of SMILE Workshop on Structure from Multiple Images in Large Scale Environments*, volume 1506, pages 154–170, 1998.

[20] R. Fleming, A. Torralba, and E. H. Adelson. How image statistics drive shape-from-texture and shape-from-specularities. In *Proc. of Third Annual Meeting of the VVS*, 2003.

[21] R. Fleming, A. Torralba, and E. H. Adelson. Specular reflections and the perception of shape. volume 4(9), pages 798–820, 2004.

[22] R. W. Fleming, R. O. Dror, and E. H. Adelson. How do humans determine reflectance properties under unknown illumination? In *Journal of Vision*, volume 3, pages 347–368, 2003.

[23] P. Giblin and R. Weiss. Reconstruction of surfaces from profiles. *Proceedings of the International Conference Computer Vision*, pages 136–144, 1986.

[24] G.Slabaugh, W. Culbertson, T. Malzbender, M. Stevens, and R. Schafer. Methods for volumetric reconstruction of visual scenes. *International Journal of Computer Vision*, 57(3):179–199, 2004.

[25] M. Halsead, A. Barsky, S. Klein, and R. Mandell. Reconstructing curved surfaces from reflection patterns using spline surface fitting normals. In *SIGGRAPH*, 1996.

[26] L. N. Hambrick, M. H. Loew, and R. L. Carroll. The entry-exit method of shadow boundary segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 9(5):597–607, September 1987.

[27] M. Hatzitheodour and M. Kender. An optimal algorithm for the derivation of shape from shadows. In *Proc. of Computer Society Conference on Computer Vision and Pattern Recognition*, pages 486–491, Ann Arbor, MI, June 1988.

[28] G. Healey and T. Binford. Local shape from specularity. *Computer Vision, Graphics, and Image Processing*, 42:62–86, 1988.

[29] B. K. P. Horn and M. J. Brooks. *Shape from Shading*. MIT Press, 1989.

[30] K. Ikeuchi. Determining surface orientation of specular surfaces by using the photometric stereo method. *IEEE Journal of Pattern Analysis and Machine Intelligence*, 3:661–669, 1981.

[31] M. Kampel, S. Tosovic, and R. Sablatnig. Octree-based fusion of shape from silhouette and shape from structured light. *Intl. Symposium on 3D Data Processing Visualization and Transmission*, pages 754–757, 2002.

[32] S. Kim, H. Hagh-Shenas, and V. Interrante. Conveying shape with texture: experimental investigations of texture's effects on shape categorization judgments. In *IEEE Transactions on Visualization and Computer Graphics*, 2004.

[33] S. K.Nayar, A. C.Sanderson, L. E.Weiss, and D. A.Simon. Specular surface inspection using structured highlight and gaussian images. *T-RA*, 6:208–218, 1990.

[34] D. Knill. Surface orientation from texture: Ideal observers, generic observers and the information content of texture cues. In *Vision Research*, volume 38, pages 1655–1682, 1998.

[35] J. Koenderink. What does the occluding contour tell us about solid shape? *Perception*, 13:321–330, 1984.

[36] J. Koenderink and A. van Doorn. Photometric invariants related to solid shape. *Optica Apta*, 27:981–996, 1980.

[37] D. Kriegman and P. Belhumeur. What shadows reveal about object structure. *Journal of the Optical Society of America - A*, 18(8):1804–1813, 2001.

[38] K. N. Kutulakos and S. M. Seitz. A theory of shape by space carving. In *Proc. of the Seventh IEEE Interational Conference on Computer Vision*, pages 307–313, Kerkyra, Greece, Sept. 1999.

[39] M. S. Langer, G. Dudek, and S. W. Zucker. Space occupancy using multiple shadow images. In *Proc. of the International Conference on Intelligent Robotics and Systems*, Pittsburg, PA, August 1995.

[40] A. Laurentini. How far 3D shapes can be understood from 2D silhouettes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(2):188–195, February 1995.

[41] S. Lazebnik, E. Boyer, and J. Ponce. On computing exact visual hulls of solids bounded by smooth surfaces. *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, pages 156–161, 2001.

[42] B. Leibe, Thad Starner, William Ribarsky, Zachary Wartell, David M. Krum, Justin Weeks, Brad Singletary, and Larry F. Hodges. Toward spontaneous interaction with the perceptive workbench. *IEEE Computer Graphics & Applications*, 20(6):54–65, November 2000.

[43] W. Lorensen and H. Cline. Marching cubes: A high resolution 3D surface construction algorithm. *Computer Graphics*, 21:163–169, 1987.

[44] R. Lu, J. Koenderink, and A. M. Kappers. Surface roughness from highlight structure. In *Applied Optics*, volume 38, pages 2886–2894, 1999.

[45] M. Goesele M. Tarini, H. Lensch and H.P. Seidel. Shape from distortion: 3d range scanning of mirroring objects. In *Proc. of SIGGRAPH, Sketches & Applications*, page 248, 2002.

[46] J. Malik and R. Rosenholtz. Surface orientation from texture: Isotropy or homogeneity (or both?). In *Vision Research*, volume 37, pages 2283–2293, 1997.

[47] W. N. Martin and J. K. Aggarwal. Volumetric descriptions of objects from multiple views. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 5(2):150–158, March 1983.

[48] S. Nayar and M. Oren. Visual appearance of matte surfaces. In *Science*, volume 267, pages 1153–1156, 1995.

[49] M. Oren and S. K.Nayar. A theory of specular surface geometry. *Trans. International Journal of Computer Vision*, pages 105–124, 1997.

[50] D. Perard. *Automated Visual Inspection of Specular Surfaces with Structured-lighting Reflection Techiniques*. PhD thesis, VDI Verlag (Nr. 869), 2001.

[51] D. Raviv, Y.-H. Pao, and K. A. Loparo. Reconstruction of three-dimensional surfaces from two-dimensional binary images. *IEEE Transactions on Robotics and Automation*, 5(5):701–710, October 1989.

[52] M. K. Reed and P.K. Allen. 3-d modeling from range imagery: An incremental method with a planning component. *Image and Vision Computing*, 17:99–111, 1999.

[53] C. Rocchini, P. Cignoni, C. Montani, P. Pingi, and R. Scopigno. A low cost optical 3D scanner. *Compute Graphics Forum*, 20(3):299–309, 2001.

[54] S. Savarese, M. Chen, and P. Perona. Second order local analysis for 3d reconstruction of specular surfaces. In *Proc. of 3D Data Processing Visualization and Trasmission*, June 2002.

[55] S. Savarese, M. Chen, and P. Perona. Recovering local shape of a mirror surface from reflection of a regular grid. In *Proc. of European Conference of Computer Vision*, 2004.

[56] S. Savarese, M. Chen, and P. Perona. Recovering local shape of a mirror surface from reflection of a regular grid. In *Proc. of European Conference of Computer Vision*, 2004.

[57] S. Savarese, L. Fei-Fei, and P. Perona. Can we see the shape of a mirror? In *Proc. of 3rd Annual Meeting of VVS*, 2003.

[58] S. Savarese and P. Perona. Local analysis for 3d reconstruction of specular surfaces. In *Proc. of IEEE Conference of Computer Vision and Pattern Recognition.*, December 2001.

[59] S. Savarese and P. Perona. Local analysis for 3d reconstruction of specular surfaces – part ii. In *Proc. of European Conference of Computer Vision*, June 2002.

[60] S. Savarese, H. Rushmeier, F. Bernardini, and P. Perona. Shadow carving. *Proc. of the Int. Conf. on Computer Vision*, 2001.

[61] S. Savarese, H. Rushmeier, F. Bernardini, and P. Perona. Implementation of a shadow carving system for shape capture. *Proc. of 3D Data Processing Visualization and Trasmission*, 2002.

[62] S. M. Seitz and C. R. Dyer. Photorealistic scene reconstruction by voxel coloring. *Proc. of Computer Society Conf. on Computer Vision and Pattern Recognition*, pages 1067– 1073, 1997.

[63] S. A. Shafer and T. Kanade. Using shadows in finding surface orientations. *Computer Vision, Graphics and Image Processing*, 22:145–176, 1983.

[64] J. Solem, H. Aans, and A. Heyden. A variational analysis of shape from specularities using sparse data. In *International Symposium on 3D Data Processing Visualization and Transmission*, 2003.

[65] S. Sullivan and J. Ponce. Automatic model construction and pose estimation from photographs using triangular splines. *IEEE Trans. Pattern Anal. Mach. Intell.*, 20(10):1091–1097, 1998.

[66] R. Swaminathan, M. D. Grossberg, and S. K. Nayar. Caustics of catadioptric cameras. In *Proc. International Conference on Computer Vision*, 2001.

[67] R. Swaminathan, S. K. Nayar, and M. D. Grossberg. Designing mirrors for catadioptric systems that minimize image errors. In *Proc. of the Fifth Workshop on Omnidirectional Vision*, 2004.

[68] R. Szeliski. Rapid octree construction from image sequences. *Computer Vision, Graphics and Image Processing*, 58(1):23–32, July 1993.

[69] R. Szeliski and P. Golland. Stereo matching with transparency and matting. *Int. Journal of Computer Vision*, 32(1):45–61, 1999.

[70] G. Taubin. A signal processing approach to fair surface design. In *SIGGRAPH*, 1995.

[71] G. Taubin, T. Zhang, and G. Golub. Optimal surface smoothing as filter design. Technical Report RC-20404, IBM Research, Mar. 1996.

[72] J. T. Todd and R. A. Akerstrom. Perception of three-dimensional form from patterns of optical texture. In *Perception and Psychophysics*, volume 13, pages 242–255, 1987.

[73] J. T. Todd, J. F. Norman, and J. Koenderink. Effects of texture, illumination, and surface reflectance on stereoscopic shape perception. In *Perception*, volume 26, pages 807–822, 1997.

[74] A. Treuille, A. Hertzmann, and S. Seitz. Example-based stereo with general BRDFs. *Proc. of the European Conference on Computer Vision*, pages 457–469, 2004.

[75] R.Y. Tsai. A versatile camera calibration technique for high accuracy 3d machine vision metrology using off-the-shelf tv cameras and lenses. *IEEE Journal of Robotics and Automation*, 3(4):323–344, August 1987.

[76] R. Vaillant and O. Faugeras. Using extremal boundaries for 3D object modeling. *IEEE Transaction of Pattern Analysis and Machine Intelligence*, 14(2):157–173, 1992.

[77] J. Wang and K. L. Dana. A novel approach for texture shape recovery. In *Proc. International Conference on Computer Vision*, 2003.

[78] Kwan-Yee Kenneth Wong. *Structure and Motion from Silhouettes*. PhD thesis, University of Cambridge, 2001.

[79] David Kwun-Moo Yang. *Shape from Darkness Under Error*. PhD thesis, Columbia University, 1996.

[80] Y. Yu and J. T. Chang. Shadow graphs and surface reconstruction. *Proc. of the European Conference on Computer Vision*, pages 31–45, 2002.

[81] Q. Zaidi and A. Li. Limitations on shape information provided by texture cues. In *Vision Research*, volume 47, pages 815–835, 2002.

[82] J. Zheng and A. Murata. Acquiring a complete 3d model from specular motion under the illumination of circular-shaped light sources. *IEEE Journal of Pattern Analysis and Machine Intelligence*, 8, 2000.

[83] J. Y. Zheng. Acquiring 3-D models from sequences of contours. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16(2):163–178, February 1994.

[84] A. Zisserman, P. Giblin, and A. Blake. The information available to a moving observer from specularities. *Image and Video Computing*, 7:38–42, 1989.